Practical Electronics

The UK's premier electronics and computing maker magazine

Circuit Surgery
Exploring the
LM35 temp sensor

Audio Out
Vocoder:
Audio PSU

Cool Beans
Mastering switch
debounce

Make it with Micromite
Build an iButton-based
Electronic Door Lock

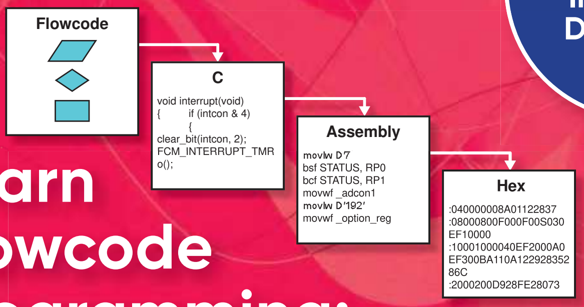
64-key MIDI Matrix

ARDUINO

High-current Battery Balancer

WIN!
Microchip MPLAB ICD 4 In-Circuit Debugger
WIN

Digital FX Unit



C

```
void interrupt(void)
{          if (intcon & 4)
           {
clear_bit(intcon, 2);
FCM_INTERRUPT_TMR
o();
```

Assembly

```
movlw D'7'
bsf STATUS, RP0
bcf STATUS, RP1
movwf _adcon1
movlw D'192'
movwf _option_reg
```

Hex

```
:040000008A01122837
:08000800F000F00S030
EF10000
:10001000040EF2000A0
EF300BA110A122928352
86C
:2000200D928FE28073
```

Learn Flowcode Programming: PIC, Arduino and 16x2 LCD

PLUS!

Techno Talk – Go eco, get ethical!
Fox Report – Another fine mess: moving to Windows 11
Net Work – Scanners, eVTOLs and the latest from space

Apr 2022 £5.49
04
9 772632 573023

www.electronpublishing.com | @practicalelec | practicalelectronics

Open the Door to IoT

Wi-Fi® Meets Open-Source Linux®

Enable Wi-Fi on any Linux-based MPU with Microchip's turn-key, pre-certified Wi-Fi modules with Mainline Linux driver support. We offer ready-to-use Linux drivers for Wi-Fi Link Controllers, ATWILC1000 (Wi-Fi only) and ATWILC3000 (Wi-Fi + BLE 5). Cut down on development time with our rich selection of resources on Github (https://github.com/linux4sam) including drivers, kernel, demo images (with driver included) and a Getting Started User Guide. We provide images for host MPUs ATSAMA5D2 and ATSAMA5D4 or you may take advantage of its flexibility and use MPUs from other vendors via the Linux mainline kernel.

Get started today with the ATWILC1000-SD evaluation kit.

- Wi-Fi IoT link controller module connects to Microchip AVR® MCUs, MPUs and other processors
- SPI and SDIO host interfaces optimized for Linux
- Supports Personal & Enterprise IEEE 802.11 WEP, WPA, WPA2 Security

For more information, visit our Getting Started page at www.microchip.com/wifi-for-linux.

MICROCHIP

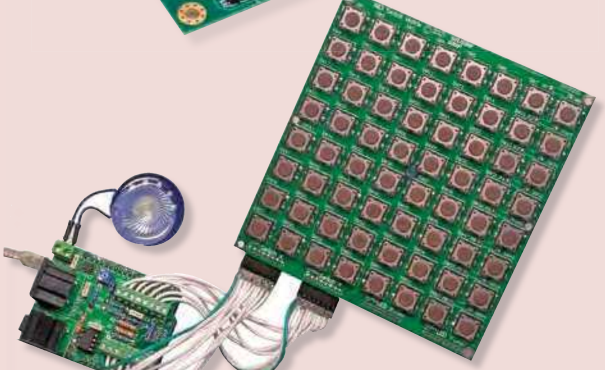
microchip.com/wifi-for-linux

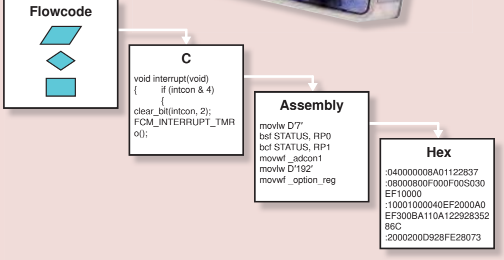
The Microchip name and logo, the Microchip logo and AVR are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. All other trademarks are the property of their registered owners.
© 2021 Microchip Technology Inc. All rights reserved.
DS70005483A. MEC2401A-UK-11-21

Practical Electronics

Volume 51. No. 4
April 2022
ISSN 2632 573X

Contents





```
void interrupt(void)
{         if (intcon & 4)
          {
clear_bit(intcon, 2);
FCM_INTERRUPT_TMR
o();
```

```
movlw D'7'
bsf STATUS, RP0
bcf STATUS, RP1
movwf _adcon1
movlw D'192'
movwf _option_reg
```

```
:040000008A01122837
:08000800F000F00S030
EF10000
:10001000040EF2000A0
EF300BA110A122928352
86C
:2000200D928FE28073
```

Made in the UK.

Written in Britain, Australia, the US and Ireland.

Read everywhere.

© Electron Publishing Limited 2022

Copyright in all drawings, photographs, articles, technical designs, software and intellectual property published in *Practical Electronics* is fully protected, and reproduction or imitation in whole or in part are expressly forbidden.

The May 2022 issue of *Practical Electronics* will be published on Thursday, 7 April 2022 – see page 72.

Projects and Circuits

64-key MIDI Matrix – Part 1 by Tim Blythman 16
This simple project turns an Arduino into an 8x8-key MIDI matrix. You can customise it via software and program it to act as a MIDI pass-through.

Digital FX Unit – Part 1 by John Clarke 26
This packed effects box will produce unique sounds from an electric guitar, bass, violin or microphone preamp, or within the effects loop of an amplifier or mixer.

High-current Battery Balancer – Part 1 by Duraid Madina 38
Proper balancing is critical for long battery life, especially for lithium-based types. Unlike many other designs, this one is efficient with little heat or waste.

Series, Features and Columns

Techno Talk by Mark Nelson 8
Go eco, get ethical!

The Fox Report by Barry Fox 10
Right to repair, Windows 11 and 1 Gig broadband

Net Work by Alan Winstanley 12
A round-up of trends emerging in image scanners and the fast-changing world of space and satellite technology.

Circuit Surgery by Ian Bell 46
The LM35 temperature sensor

Flowcode Graphical Programming by Martin Whitlock 49
Part 3: LCD temperature monitor

Audio Out by Jake Rothman 56
Analogue Vocoder – Part 6: Universal Audio PSU

Max's Cool Beans by Max The Magnificent 62
Flashing LEDs and drooling engineers – Part 26

Make it with Micromite by Phil Boyce 66
Part 35: Mini-Project: iButton-controlled Electronic Door Lock

Regulars and Services

Wireless for the Warrior 2

Subscribe to Practical Electronics and save money 4

Reader services – Editorial and Advertising Departments 7

Editorial 7
It's all about power

NEW! **Practical Electronics back issues** DOWNLOADS **– 2021 now available!** 14

Exclusive Microchip reader offer 15
Win a Microchip MPLAB ICD 4 In-Circuit Debugger

PE Teach-In 8 55

Practical Electronics PCB Service 70
PCBs for Practical Electronics projects

Classified ads and Advertiser index 71

Next month! – highlights of our next issue of Practical Electronics 72

WIRELESS FOR THE WARRIOR

by LOUIS MEULSTEE

THE DEFINITIVE TECHNICAL HISTORY OF RADIO COMMUNICATION EQUIPMENT IN THE BRITISH ARMY

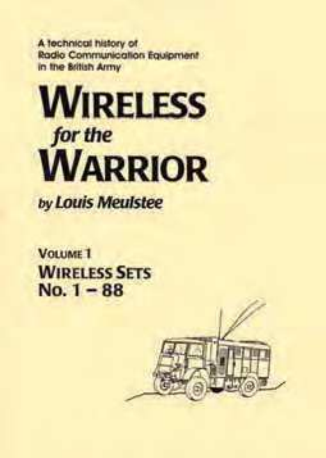
The *Wireless for the Warrior* books are a source of reference for the history and development of radio communication equipment used by the British Army from the very early days of wireless up to the 1960s.

The books are very detailed and include circuit diagrams, technical specifications and alignment data, technical development history, complete station lists and vehicle fitting instructions.

Volume 1 and *Volume 2* cover transmitters and transceivers used between 1932-1948. An era that starts with positive steps taken to formulate and develop a new series of wireless sets that offered great improvements over obsolete World War I pattern equipment. The other end of this timeframe saw the introduction of VHF FM and hermetically sealed equipment.

Volume 3 covers army receivers from 1932 to the late 1960s. The book not only describes receivers specifically designed for the British Army, but also the Royal Navy and RAF. Also covered: special receivers, direction finding receivers, Canadian and Australian Army receivers, commercial receivers adopted by the Army, and Army Welfare broadcast receivers.

Volume 4 covers clandestine, agent or 'spy' radio equipment, sets which were used by special forces, partisans, resistance, 'stay behind' organisations, Australian Coast Watchers and the diplomatic service. Plus, selected associated power sources, RDF and intercept receivers, bugs and radar beacons.

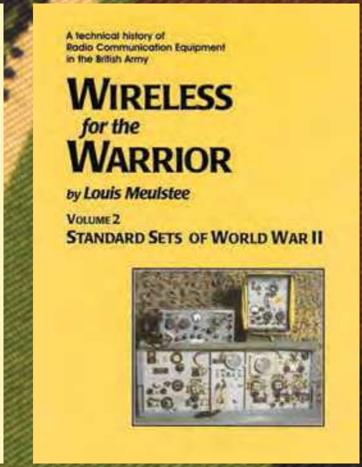


A technical history of
Radio Communication Equipment
in the British Army

WIRELESS
for the
WARRIOR

by Louis Meulstee

VOLUME 1
WIRELESS SETS
No. 1 – 88

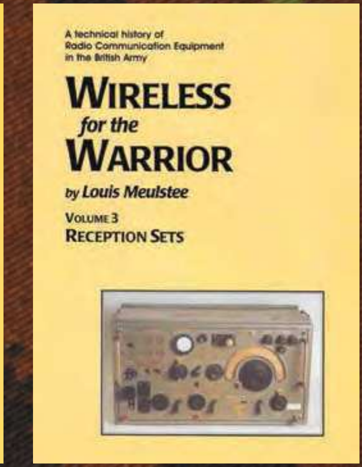


A technical history of
Radio Communication Equipment
in the British Army

WIRELESS
for the
WARRIOR

by Louis Meulstee

VOLUME 2
STANDARD SETS OF WORLD WAR II

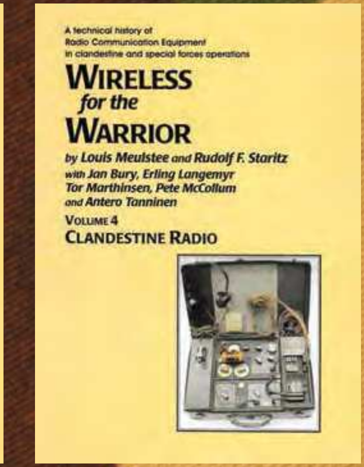


A technical history of
Radio Communication Equipment
in the British Army

WIRELESS
for the
WARRIOR

by Louis Meulstee

VOLUME 3
RECEPTION SETS



A technical history of
Radio Communication Equipment
in clandestine and special forces operations

WIRELESS
for the
WARRIOR

by Louis Meulstee and Rudolf F. Staritz
with Jan Bury, Erling Langemyr
Tor Marthinsen, Pete McCollum
and Antero Tanninen

VOLUME 4
CLANDESTINE RADIO

ORDER YOURS TODAY!
JUST CALL 01202 880299 OR VISIT www.electronpublishing.com

Quasar Electronics Limited
PO Box 6935, Bishops Stortford
CM23 4WP, United Kingdom

Tel: 01279 467799
E-mail: sales@quasarelectronics.co.uk
Web: quasarelectronics.co.uk

All prices include 20% VAT. Free UK mainland delivery on orders over £60.
Postage & Packing Options (Up to 1Kg gross weight): **UK Standard 2-5 Day Delivery** - £4.95 : **UK Mainland Next Day Delivery** - £9.95 : Please order online if you reside outside the UK (our website will calculate postage for you).
Payment: We accept all major credit/debit cards. Make UK cheques/PO's payable to Quasar Electronics Limited and include P&P detailed above.
!! Order online for reduced price postage and fast despatch !!
Please visit our online shop now for full details of over 1000 electronic kits, projects, modules and publications. Discounts for bulk quantities.

QuasarElectronics.co.uk

Solutions for Home, Education & Industry Since 1993

Official Main Dealer stocking the full range of Kits, Modules, Robots, Instruments, Tools and much, much more...

Electronic Kits & Modules

We have a massive selection of self-assembly electronic kits and pre-assembled modules. Please see the full range on our website or call for details.

LED Buddy / LED Tester Kit

Hold any type of LED to the contact pads to see it's polarity, forward voltage & the recommended series resistor value. Adjustable target current & forward voltage. Great design aid.
*Order Code: WSMI198 - **£16.66***

5-in-1 Mechanical Coding Robot Kit

Mission-based, entry level STEM (Science - Technology - Engineering - Mathematics) coding robot designed to inspire users to learn about mechanical engineering & coding basics. Solderless assembly. 2x AAA battery powered (not included). Aged 14+.
Order Code: KSR19 - ~~£32.95~~ £29.95

Stereo Ultrasonic Bat Detector Kit

This exciting, educational & fun kit converts high frequency sounds (20 - 90kHz) normally imperceptible to humans like bat signals into audible noise. Can also help detect failures in machines, engines, etc. Stereo feature adds the possibility to pinpoint the source. Frequency range 20-90kHz. 3.5mm jack output. 129x60x40mm. Requires 3x AA batteries and stereo headphones (not included).
*Order Code: K8118 - **£20.39***

Bidirectional DC Motor Speed Controller

Control the speed of most common DC motors (rated up to 32Vdc/5A) in both the forward and reverse directions. The range of control is from fully OFF to fully ON in both directions. The direction and speed are controlled using a single potentiometer. Screw terminal block for connections. PCB: 90x42mm. **Not suitable for use with lead acid batteries!**
*Kit Order Code: 3166KT - **£19.96***
*Assembled Order Code: AS3166 - **£29.95***

3 Channel RGB LED Light Organ Kit

3 outputs react to different sound frequencies. On-board microphone picks up surrounding sound or music and drives the low, mid & high frequency outputs. Connect a 12-24Vdc RGB LED strip or separate colour LED strips or LEDs (not included). Master & separate channel sensitivity adjustment. 1.25 A max. per channel. Panel mountable facia 127 x 44mm. Terminal block or jack power supply connection. Control knobs included.
*Order Code: WSL209 - **£14.10***

Stereo Preamplifier with Tone Controls

Hi-fi quality stereo preamplifier board with very pleasing performance. Volume control with integrated power switch. +/-15 dB Bass and Treble tone controls. Channel balance control. RCA jacks f or audio inputs. Up to 20 dB of voltage gain. Selectable loudness function. Philips TDA1524A based circuit.
*Kit Order Code: 3100KT - **£23.95***
*Assembled Order Code: AS3100 - **£32.95***

The Lady Soldering Kit

Meet the Lady! This self-assembly soldering kit includes both through-hole and SMD components, LEDs plus a beautiful red and white PCB.
*Order Code: WSAK206 - **£8.99***

Card Sales & Enquiries 01279 467799

Digitally Controlled FM Radio Kit

Build your own modern, high quality FM receiver project with excellent sensitivity powered by a simple 9V PP3 battery (not included. Auto-seeking button. 4 station presets. Volume control. Excellent learning project for schools and colleges.
*Order Code: MK194N - **£20.39***

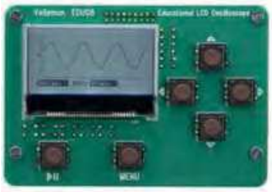
Audio Analyser Display Kit

Small, compact LCD display, ideal for panel mounting. Give your homemade audio gear a high-tech look. Upgrade existing equipment. Provides Peak Power, RMS Power, Mean dB, Peak dB, Linear Audio Spectrum And 1/3 Octave Audio Spectrum. Auto / Manual range selection. Peak-hold function. Speaker impedance selection.
*Order Code: K8098 - **£38.39***

Electronic Component Tester Kit

Build your own versatile component tester. Shows value and pin layout information for resistors (0.1 Ohm resolution, max. 50 MOhm), coils (0.01mH - 20H), capacitors (28pF - 100mF), diodes, BJT, JFET, E-IGBT, D-IGBT, E-MOS & D-MOS.
*Order Code: K8115 - **£44.15***

LCD Oscilloscope Educational Kit



Build your own LCD oscilloscope with this exciting new kit. Learn how to read signals. See the electronic signals you learn about displayed on your own LCD oscilloscope. Despite the low cost, this oscilloscope kit has a lot of features found on expensive units like signal markers, frequency, dB, true RMS readouts and more. A powerful auto-setup function will get you going in a flash!
*Order Code: EDU08 - **£48.54***

Practical Electronics

UK readers SAVE £1 on every issue

SUBSCRIBE NOW! 01202 087631

Take out a one-year subscription and save more than £10 over the year.
Even better – save over £1 per issue if you subscribe for two years – a total saving of £26.

Overseas rates represent exceptional value

You also:
- **Avoid any cover price increase for the duration of your subscription**
- **Get your magazine delivered to your door each month**
- **Guarantee a copy, even if your local newsagent sells out**

Order by:
- **Phone or post with a cheque, postal order or credit card**
- **Or order online at: www.electronpublishing.com**

Practical Electronics

SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in:

UK: 6-months £29.99, 12-months £54.99, 24-months £104.99

Europe Airmail: 6-months £33.99, 12-months £63.99, 24-months £119.99

Rest Of The World Airmail: 6-months £41.99, 12-months £77.99, 24-months £149.99

Cheques (in **£ sterling only**) payable to *Practical Electronics* and sent to: Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom

Tel: 01202 087631
Email: pesubs@selectps.com

Also via our online shop at**: www.electronpublishing.com**

Subscriptions start with the next available issue. We accept MasterCard or Visa.
(For past issues see the Back Issues page.)

DIGITAL SUBSCRIPTIONS

Electronic subscriptions are available from £24.99 for 12 months, more details at: **www.electronpublishing.com**

SUBSCRIPTION ORDER FORM

☐ **6 Months: UK £29.99, Europe £33.99 (Airmail), Rest of the World £41.99 (Airmail)**
☐ **1 Year: UK £54.99, Europe £63.99 (Airmail), Rest of the World £77.99 (Airmail)**
☐ **2 Years: UK £104.99, Europe £119.99 (Airmail), Rest of the World £149.99 (Airmail)**

To: Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH
United Kingdom

Tel: 01202 087631
Email: pesubs@selectps.com

I enclose payment of £ ....
(cheque/PO in £ sterling only)
payable to *Practical Electronics*

☐ Please charge my Visa/Mastercard

My card number is: ....
Please print clearly, and check that you have the number correct

Card Security Code .... Valid From Date........................
(The last 3 digits on or just under the signature strip)

Card Ex. Date ....

Name ....

Address ....

Post code .... Tel. ....

Email ....

Subscriptions start with the next available issue.

PicoLog Cloud®

View live and saved captures
from PicoLog data loggers
anywhere in the world,
on any device, for **FREE!**

Learn more at
www.picolog.com

Visit **www.picotech.com/A1066** to find out more

Email: sales@picotech.com. Errors and omissions excepted.
Please contact Pico Technology for the latest prices before ordering.

pico®
Technology

Subscribe to SILICON CHIP

SILICON CHIP
DECEMBER 2021
The small but powerful Hummingbird Amplifier
Hands-on with the Raspberry Pi Pico
The smallest Raspberry Pi yet!

Australia's top electronics magazine

SILICON CHIP is one of the best DIY electronics magazines in the world. Each month is filled with a variety of projects that you can build yourself, along with features on a wide range of topics from in-depth electronics articles to general tech overviews.

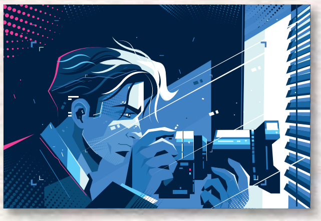
	Print	**Combined**	**Online**
6 months	$100 ≈£54 ≈€64	$110 ≈£60 ≈€70	$50 ≈£27 ≈€31
1 year	$195 ≈£105 ≈€124	$215 ≈£116 ≈€137	$95 ≈£51 ≈€60
2 years	$380 ≈£205 ≈€242	$415 ≈£224 ≈€264	$185 ≈£100 ≈€118

- *Combined subscriptions include both the printed magazine and online access.*
- *Prices are in Australian dollars (AUD) unless otherwise specified.*
- *Prices listed are for everywhere in the world except Australia and New Zealand. View siliconchip.com.au/Shop/SubRates for a full list of current prices.*
- *Price estimates are just that and should only be used as a guide, we primarily take payment in the form of Australian dollars.*

Try our Online Subscription – now with PDF downloads!

An online issue is perfect for those who don't want too much clutter around the house and is the same price worldwide. Issues can be viewed online, or downloaded as a PDF.

Published in SILICON CHIP



Big Brother is Tracking You; November 2021

USB Cable Tester; November 2021

Hummingbird Amp; December 2021

Raspberry Pi Pico; December 2021

To start your subscription go to siliconchip.com.au/Shop/Subscribe

Practical
Electronics

Volume 51. No. 4
April 2022
ISSN 2632 573X

Editorial

Editorial offices
Practical Electronics **Tel** 01273 777619
Electron Publishing Limited **Mob** 07973 518682
1 Buckingham Road **Fax** 01202 843233
Brighton **Email** pe@electronpublishing.com
East Sussex BN1 3RA **Web** www.electronpublishing.com

Advertisement offices
Practical Electronics Adverts **Tel** 01273 777619
1 Buckingham Road **Mob** 07973 518682
Brighton **Email** pe@electronpublishing.com
East Sussex BN1 3RA

Editor Matt Pulzer
General Manager Louisa Pulzer
Digital subscriptions Stewart Kearn **Tel** 01202 880299
Online Editor Alan Winstanley
Web Systems Kris Thain
Publisher Matt Pulzer

Print subscriptions
Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH **Tel** 01202 087631
United Kingdom **Email** pesubs@selectps.com

Technical enquiries

We regret technical enquiries cannot be answered over the telephone. We are unable to offer any advice on the use, purchase, repair or modification of commercial equipment or the incorporation or modification of designs published in the magazine. We cannot provide data or answer queries on articles or projects that are more than five years old.

Questions about articles or projects should be sent to the editor by email: pe@electronpublishing.com

Projects and circuits

All reasonable precautions are taken to ensure that the advice and data given to readers is reliable. We cannot, however, guarantee it and we cannot accept legal responsibility for it.

A number of projects and circuits published in *Practical Electronics* employ voltages that can be lethal. You should not build, test, modify or renovate any item of mains-powered equipment unless you fully understand the safety aspects involved and you use an RCD (GFCI) adaptor.

Component supplies

We do not supply electronic components or kits for building the projects featured, these can be supplied by advertisers. We advise readers to check that all parts are still available before commencing any project in a back-dated issue.

Advertisements

Although the proprietors and staff of *Practical Electronics* take reasonable precautions to protect the interests of readers by ensuring as far as practicable that advertisements are bona fide, the magazine and its publishers cannot give any undertakings in respect of statements or claims made by advertisers, whether these advertisements are printed as part of the magazine, or in inserts. The Publishers regret that under no circumstances will the magazine accept liability for non-receipt of goods ordered, or for late delivery, or for faults in manufacture.

Transmitters/bugs/telephone equipment

We advise readers that certain items of radio transmitting and telephone equipment which may be advertised in our pages cannot be legally used in the UK. Readers should check the law before buying any transmitting or telephone equipment, as a fine, confiscation of equipment and/or imprisonment can result from illegal use or ownership. The laws vary from country to country; readers should check local laws.

It's all about power

Jake Rothman's opening line in this month's *Audio Out* column – 'Audio lives or dies by its power supplies…' – is of course aimed at audio electronics fans, but it's a good reminder that all our projects, designs and explorations in the world of electronics come to nothing if the battery, PSU, supply or whatever source of electrical power you are using isn't up to the job.

We do run plenty of power supply projects, but I hope you'll take the time to read through Jake's latest design. Even if audio is not your main focus, it's an excellent example of elegant analogue design that could be used in many other applications that need stable power with low levels of ripple and hum. As always, Jake's designs use inexpensive, readily available components to get excellent results that anyone can build.

Advancing with Flowcode

Martin Whitlock returns in this issue with another in-depth exploration of how to use Flowcode. You'll recall from his January and February articles that Flowcode is a flexible and intuitive programming environment that lets you program a PIC16F88, Arduino Uno and many other PICs/microcontrollers without having to use assembler or C – you just draw flowcharts of the intelligent process that you want to implement. This month, he shows you how to create a simple electronic thermometer and tackles two common requirements for microcontroller users – reading an analogue voltage and writing to a 16x2 character display.

A nice piece of serendipity is that this month's *Circuit Surgery* looks at the LM35 temperature sensor IC. This is exactly the same device used in the Flowcode mini project, so there's analogue and digital focus on one useful and easy-to-follow project.

Projects, projects, projects

Speaking of projects, in addition to our regular columns, this month we start three fascinating designs for you to build – a useful *High Current Battery Balancer*, a handy *64-key MIDI Matrix* and a top-quality *Digital FX Unit*. Plus, Phil Boyce explains all you need to know to build a Micromite-based *iButton Electronic Door Lock*.

Lots to keep you busy!

Keep well and stay healthy

Matt Pulzer
Publisher

Go eco, get ethical!

Techno Talk

Mark Nelson

These days, we hear a lot about 'mindfulness', which according to one website is the basic human ability to be fully aware of where we are and what we're doing. The website doesn't mention behaving responsibly or paying attention to what's going on around us, but maybe if more people did this, the world would be a better place.

One company that clearly has awareness is Tesco, whose slogan 'Every little helps' expresses the company's approach to sustainability when you shop at their stores. Thinking about electronics, another company to which the slogan could equally apply is Samsung Electronics, whose PlanetFirst eco-management system offers customers environmentally conscious solutions that lead the way to a sustainable future. At this year's CES trade show in Las Vegas, billed as, 'the most influential tech event in the world,' Samsung launched its upgraded Eco Remote controller for television sets and other home appliances. When previewed last year, it was a solar-powered device that the manufacturer estimated could prevent 99,000,000 AAA batteries going to landfill over the next seven years. This year's version can additionally recharge without daylight by harvesting energy from your domestic Wi-Fi router at distances of up to 40m.

Battery-free but not pointless

A spokesperson for Samsung declared that whether it's a bright and sunny day or the middle of the night, the battery stays fully charged by collecting radio waves from Wi-Fi networks and converting them to energy. 'There is no lithium-ion battery in the remote,' continued the representative. 'What it does have is a capacitor that can store small amounts of energy.'

Endorsing the practicality of the principle, Gus Cheng Zhang, a lecturer in power electronics at the University of Manchester, explained that most of the energy radiated by Wi-Fi routers was usually wasted, absorbed by metallic surfaces, creating eddy currents and then turned into heat. He told *New Scientist* magazine: 'It's definitely a brilliant idea, and a very good example of energy harvesting. You have a router that is emitting radio frequency signals that's rated at approximately 1W. This is limited by regulations and that's mostly just concerns about safety. There is no technical limitation to how much power can be transmitted from the router to a device.' He concluded that other practical applications existed for the technology in certain niches, although solar power offered similar output for devices with low power needs at less cost.

Eco and ethics

Ecology, environmental sustainability and ethics are all closely interrelated, but exactly how is a matter of debate. Some people argue that a sense of duty to nature may be beneficial, but it's hardly a reliable way to achieve sustainability. On the other hand, the fact that Amazon now offers its Amazon Renewed range of refurbished, tested and guaranteed products is definitely a step in the right direction. Another sign of progress was the staging of the first World Ethical Electronics Forum (WEEF) in Germany last year, offering global innovators in electronics an open discussion about ethics and sustainable development goals (SDGs). These are early days and the second WEEF to be held in November should present a far clearer identification of best practice in ethical electronics and provide answers to questions such as whether individual designers, programmers and makers have an ethical responsibility to use their skills and components in positive ways. Also, should ethics play a role in how multinational electronics-focused corporations do business?

Let's loosen licensing

Friendlier licensing was one of the subjects discussed at last year's WEEF conference and caught my eye as something highly relevant to practical electronicists like us. The German software developer Segger Microcontroller chaired a discussion on the novel subject of 'friendly licensing', which allows anyone to use high-end software free of charge for non-commercial use and for testing purposes. Segger's marketing manager Frank Riemenschneider briefed *Elektronik Praxis* magazine on why and how this should work, saying: 'The more that young people have the opportunity to develop innovative and energy-saving devices, the sooner we will achieve global climate-neutrality. To make the world a better place, we need knowhow for forward-looking projects. This is especially true for software needed for training and tinkerers alike.

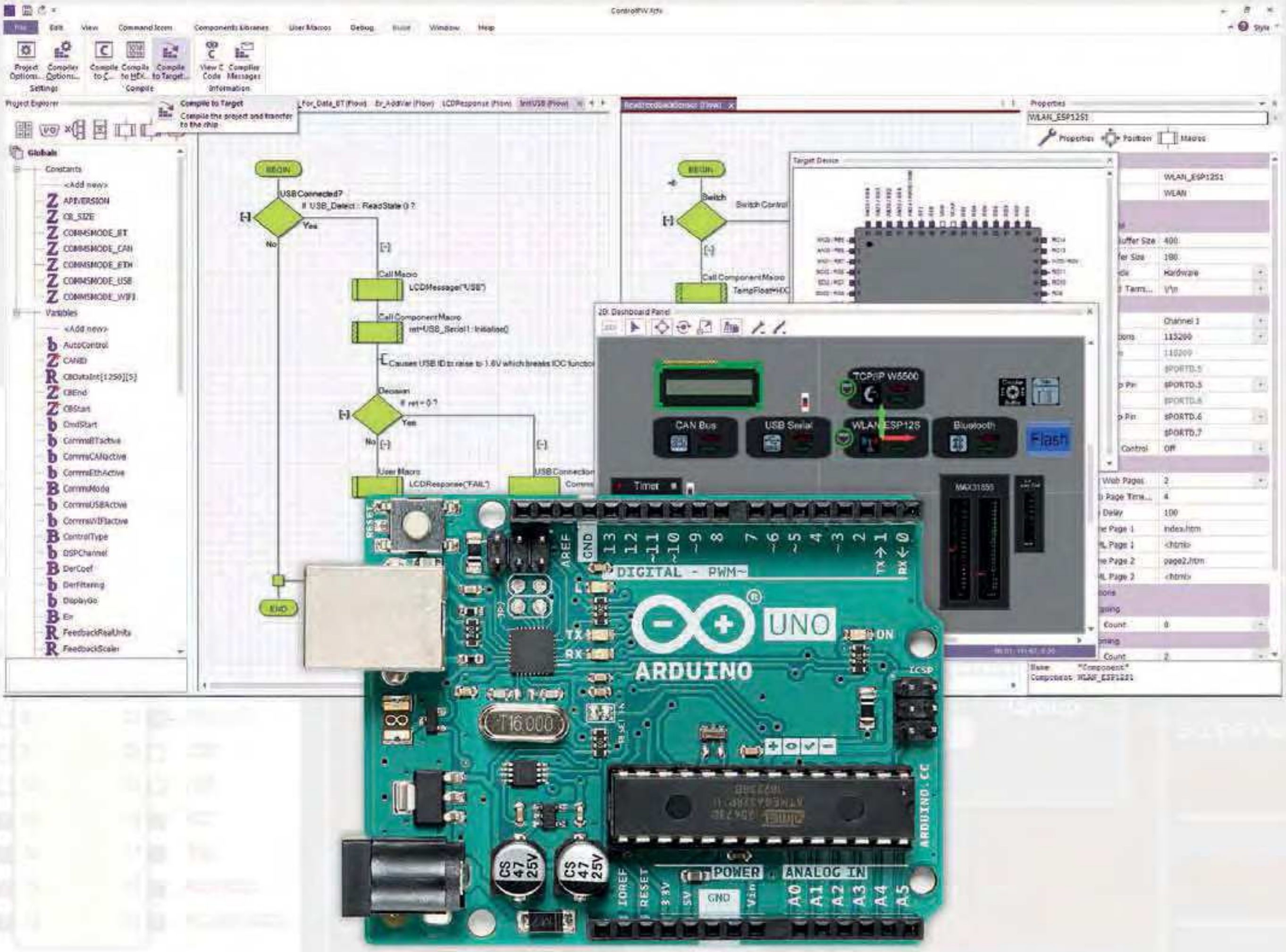
'Unfortunately, not all schools and universities in the world – and not all students or simply curious, technically interested young people – can afford expensive software that they would need for a good education. For these people, we have introduced our Friendly Licensing model, so that everyone can learn with state-of-the-art software packages, regardless of their financial circumstances.

'For non-commercial use or evaluation purposes, no matter who you are, you can use the software free of charge under this licence. Non-commercial use means that you use the software for teaching, learning, studying or for projects that have no commercial background, and evaluation means that you try out the software for possible future use. In practice, students, tutors, teachers or individuals can use our software in universities, colleges, non-profit organisations or at home. Hobbyists can use it for projects, courses and self-education.'

Benefits all round

Do note that Segger's software is not shareware, trialware or demoware. You might perhaps call it 'fairware', except that several businesses and organisations have already grabbed the name. The software company retains full copyright of its product and professional users must still pay to use it. But students and hobbyists, who could not afford to buy the programs, get to use it for nothing.

Another company that has adopted this model is Blackmagicdesign, whose Davinci Resolve video and audio post-production software is used both on Hollywood feature films such as *Deadpool 2*, *La La Land*, *Star Wars: The Last Jedi* and *X-Men: Apocalypse*, as well as on television shows such as *The Muppets*, *Game of Thrones* and *Modern Family*. All of these users pay for the commercial edition, while more than two million non-professionals around the world use the free edition, which has slightly reduced functionality. Friendly licensing really does work!



CREATE FANTASTIC ELECTRONIC SYSTEMS USING

FLOWCODE

20% off your Flowcode purchase using code: EPE20

Use code at checkout: flowcode.co.uk/buy

The Fox Report

Barry Fox's technology column

Right to repair, Windows 11 and 1 Gig broadband

In the US, President Biden has signed an Executive Order which tells the Federal Trade Commission (FTC) to pass Right to Repair laws. These will force manufacturers to let individuals and independent repair companies try to fix broken equipment rather than junk and replace. The intentions are admirable, but there is a good chance of tears before bedtime as companies introduce fanciful policies.

Witness the new 'R for Repair' tag unveiled by Dutch Philips and Chinese partner TP Vision at a recent online press conference. I tried to re-watch the live event and so be sure I had been hearing and seeing correctly. But it was gone, apparently for ever. My memory and notes are clear, though. Philips/TP Vision is now promising to let owners of TV sets do DIY repairs.

Several journalists who were logged into the event, including myself, asked for more information on how Joe Public or a handyman-for-hire can get inside a modern TV set and repair faults. The answer we got from TP Vision managers was vague in the extreme. 'Obviously people will not be able to replace components that are glued together (sic). But they will be able to do some basic things. We are on a journey. We are taking it step by step'.

We asked for examples of 'basic things', but got none.

If anyone is capable of identifying which microchip on the motherboard of a TV has gone bad, and then desoldering and replacing it with a new one obtained from who-knows-where, it is most likely to be a *PE* reader. Less tech-savvy owners are more likely to try hitting it with a hammer or dolloping in some WD40… inevitably finishing up with duct tape.

Another fine mess

Owners of Windows PCs have for many months been receiving alerts from Microsoft on Windows 11, the free upgrade to Windows 10 which (allegedly) makes a Windows PC look and feel more like a Mac.

The upgrades have been staggered, by location and hardware, and who knows what other criteria, but all the PCs I have direct contact with have now been offered the chance to go up to 11. Already, there are lessons to be learned.

Because the 11 upgrade offer arrives with other updates, such as security improvements, one owner clicked 'Yes' to 11 without realising the significance. The download then started, installed and prepared the PC for a re-start. We could find no risk-free way to undo the install. So, it duly upgraded to Windows 11.

Three other machines all declined to upgrade, blaming inadequate hardware. Downloading the free software tool Health Check (which Microsoft offers via its ordinary Update option) revealed that each PC had adequate HD space and RAM, but all the CPUs were deemed inadequate even though they met the overall speed requirements. An Intel Pentium, Intel i7 and AMD A10-6700 were all deemed unworthy of Windows 11.

Whole rafts of relatively recent and respected processors are also excluded. So, it is likely that many readers – and many businesses – will now find their systems rejected for upgrade to Windows 11.

There are two options. Try and bypass the check by clean-installing Windows 11, or simply stay with Windows 10.

Don't force Windows 11

I'd strongly advise against trying to force an install. Although conspiracy theorists will doubtless see the chip rejection as part of some grand plan

HAMMOND

new!

1551V snap-fit vented and plain miniature plastic enclosures

Learn more: hammfg.com/1551v

More than 5000 standard stocked enclosure designs
uksales@hammfg.com • 01256 812812

to sell new PCs, the length of the 'No-to-11' list makes this highly implausible. Declined Windows 10 users are more likely to buy into the Mac world or experiment with Linux or Android Chromebooks, thereby achieving the exact opposite of what Microsoft would want.

Everything points to the fact that Microsoft has blocked routine upgrades from 10 to 11 because pre-launch testing has thrown up serious crash and compatibility problems. So be very wary of trying to over-ride the block.

Bugs, patches and keys

Also, it is unclear whether and how a forced upgrade to 11 will receive and cope with the essential bug fixes and security patches to Windows 11 that are sure to follow. As I found out recently when a routine Windows 10 update somehow de-activated the key for a new version of Microsoft Office and replaced it with an expired key for an old trial version, juggling keys in the Windows registry requires a brave heart, strong nerve and fair knowledge of DOS Command line working; as well as self-preservation creation of registry back-ups and system restore points, in case anything goes wrong.

Your starter for 10

The other, far better option is to stick with Windows 10. Microsoft has assured that support with updates will continue until October 2025. It's likely that by then Microsoft will have patched and tweaked Windows 11 to handle many of the currently vetoed processors; and in the meantime, fixed any number of other bugs that first-time users of Windows 11 will have suffered.

Just one example: The biggest criticism of ill-fated Windows 8 was that it removed the familiar Start menu, just as Windows users had grown to ignore the absurdity of hiding the Off or Shut Down option under a Start label. A small industry grew out of third-party software such as StartIsBack that cosmetically modified the unfriendly and unfamiliar Windows 8 user interface so that it looked like the familiar and friendly Windows 7 screen.

With a change of top management, Microsoft skipped Windows 9 and launched Windows 10, with the friendly, previously familiar Start menu re-instated.

A colleague whose PC recently updated from 10 to 11 immediately found that the Start Option to Stop had disappeared; in desperation they Shut Down by physically pressing the ON/Off switch.

Start has not actually been removed. It has just been hidden in the middle of a different-looking task bar. A few easy-when-you-know-how mouse clicks on hidden menu options will send the Start button back to where users expect it to be.

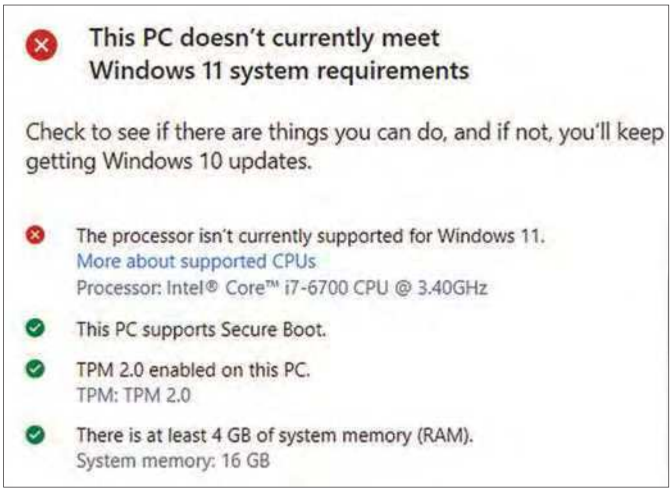
A new generation of StartIsBack and similar third-party software is already available for Windows 11.

What we-know-better-than-our-users mastermind inside Microsoft thought it would be a good idea to repeat the Windows 8 user alienation trick? Will an update to 11 now replace the Start button?

Giga-battle

We are now seeing the start of a new broadband battle, with 1 Gig FTTP (1Gbps from fibre-to-the-premises) at the heart of the sales pitch.

Doubtless spurred on by new entrants to the market, like G Network in London, Virgin says it is now 1-Gig-capable across its full network (subject to availability and network capacity, whatever that means) – see: **www.virginmedia.com/broadband/gigabit**


This PC doesn't currently meet Windows 11 system requirements

Check to see if there are things you can do, and if not, you'll keep getting Windows 10 updates.

The processor isn't currently supported for Windows 11.
More about supported CPUs
Processor: Intel® Core™ i7-6700 CPU @ 3.40GHz

This PC supports Secure Boot.

TPM 2.0 enabled on this PC.
TPM: TPM 2.0

There is at least 4 GB of system memory (RAM).
System memory: 16 GB

This Windows 11-rejection message is frustrating for owners of Windows 10 (or earlier) machines, but proceed very carefully if you choose to ignore it and force an upgrade.

Virgin is being upfront about the elephant in room – only a few people with big houses and big needs actually need 1 Gig capacity.

'Whether you've got a full house, are into serious gaming, or need to share some hefty files, 1 Gig1 Fibre Broadband has enough speed to please even the busiest of households,' says Virgin. 'Big family? Gig1 Fibre Broadband can power it all at the same time, without breaking a sweat. Stream Ultra HD videos while downloading mammoth files, and your kids can still FaceTime their mates. Plug in your console and game on. There's zero traffic management, ever, so your connection won't slow down even during peak times. Big house share? End the battle for the bandwidth. Gig1 Fibre Broadband's got plenty of speed for everybody. You can all stream in Ultra HD, game, host Zoom calls in different rooms, download and upload university work, and more – all at the same time.'

Which, when translated, means that FTTP will be expensive overkill for Mr or Ms Average Internet user.

Is it really that fast?

There is another elephant in the room. Although 1 Gig may be coming into a house, speeds are likely to drop dramatically when the signal is sent round the house. Here's what a tech savvy friend reports from his first experience with FTTP from G Network in London.

Speeds 'down' into the home, and 'up' out of the home are, as is claimed, over 900 Mbps and nearly the aspirational 1 gigabit. But, and it's a big but, this only holds good if the broadband signal is fed direct from the fibre router into a computer by a short Ethernet cable. As soon as the broadband signal is sent round the home by a Wi-Fi system, even one labelled '1Gb', the real speed drops to around 400Mbps.

My user friend writes: 'Speeds of 900Mbs plus are nice and something to brag about. But I enjoyed 100Mbs previously with Virgin and that was more than fast enough for me, so (rhetorical question) who needs nearly a gigabit residentially? Since they offered me a year free as 'the first person in the house to take up the offer' who am I to refuse? If it all goes sideways the Virgin cabling is still in place, so reverting would be painless. Watch this space!'

Net Work

Alan Winstanley

This month's *Net Work* brings readers a round-up of trends emerging in image scanners and the fast-changing world of space and satellite technology.

These days, images and video are easily captured using remarkably high-resolution smartphones and digital cameras, and images can be hosted either locally or on the cloud. Occasionally though, legacy artwork, photos or films crop up that need digitising – faced with piles of prints, 35mm slides or strips of negatives, a flatbed scanner with a transparency adaptor is the obvious answer.

Some readers doubtless remember the 'three pass' scanner (one pass per RGB colour), which operated painfully slowly via a PC parallel port or a 'fast' SCSI card. 30 years on, and a new addition arrived in my worklab in the form of a flatbed USB film scanner. Nothing out of the ordinary, but research had showed that flatbed scanners had gone the same way as webcams in terms of price, support and availability. There's a dearth of available flatbed film scanners, and prices for what seem like long-in-the-tooth legacy units were high – if you could find one at all, as some mainstream makers have simply stopped making them.

There's clearly some demand because used ones fetch silly prices on eBay, so I reluctantly paid a premium for a highly regarded Epson V600 Photo scanner because cheaper models like the V550 weren't available at all. Of course, more time is then wasted figuring out differences in features and wondering what more you actually get for an extra 50% in price (the answer is mainly the software bundle and some infrared specs – see later). The Epson V600 scanner has a transparency backlight in the lid and is targeted at semi-pro home users. The Camelizer plug-in on **Amazon.co.uk** revealed this model had been on the go for over ten years; hopefully, it's been updated over time, but prices had gone up approximately 50% over the duration and are touching £300 (or more) today (The similar but unavailable V550 was £200). So, this set some alarm bells ringing: if it was launched in the era of Windows 7 or 8, how would it fare today under Windows 10 or even Windows 11?

Regular PC users know that installing add-ons like scanners usually takes either ten minutes or ten hours. Epson clearly stated that it was W10 and W11 compatible, but my own research indicated what to expect if installing one on a modern PC. Customer reviews and forum posts revealed that several users had complained of software lock-ups under Windows 10 and eventually some deep links to updated drivers were found: in Epson's case, these should be installed before connecting the scanner, so I downloaded them in advance, ready for when the scanner arrived.

Back to the future

My installation initially went perfectly under Windows 10, despite having an Epson all-in-one printer also installed on the same PC. I found I could select the V600 scanner in both the 32-bit and 64-bit versions of PaintShop Pro. (Therein lies another undocumented problem: legacy scanner TWAIN drivers, as well as expensive plug-in graphics filters, are often only compatible with 32-bit graphics software. Happily, PaintShop Pro provides both 32-bit and 64-bit versions for backwards compatibility: I run both versions on the same PC mainly for that reason.)

The scanner is fast and accurate but, disappointingly, some of the scanner's Help resources were very dated HTML pages that looked like a throwback to Windows XP, and unsurprisingly the software instructions did not cover the entirely new version I'd tried under Windows 10. Oddly, some major documented scanner features and functions were missing altogether. One of them was Epson's highly regarded Digital Image Correction and Enhancement (ICE), which can help remove 'stubborn dust marks' on some types of film. Such ICE-enabled scanners contain infrared hardware to highlight specks of dust and scratches 'invisibly' and digitally erase them from the raw image, which is why you buy them: the Digital ICE option was not available in the latest Epson driver. The familiar old 'Home' or 'Professional' setting modes weren't there either.

Epson's film-capable mid-range V600 Photo scanner with transparency unit. Film scanner choices are becoming limited as users migrate over to digital media.

Kodak's Mini Digital Film and Slide Scanner will digitise images at the press of a button and store them on a memory card. (Amazon.co.uk)

Eating into those ten hours, more trials with legacy drivers eventually cured the problem, for now at least. Although the supplied 'Epson Scan' software on CD was showing its age, it did include Digital ICE options. It seems the latest 'Scan 2' update does not. Full Auto, Home and Professional modes were restored on my W10 setup too. The software's dated, but it works, though the signs are less promising for Windows 11 upgraders. One forum user stated that Epson offered them a lite version of professional Silverfast software (**www.silverfast.com**) as it includes iSRD (infrared-based Dust & Scratch Removal). If ICE is a 'must-have' then Silverfast SE will set you back another €49. For many flatbed scanner owners though, the go-to source for universally compatible and sophisticated scanner software is undoubtedly VueScan. Its features include 'infrared cleaning' and it can drive nearly 6,000 scanners, even obsolete ones. VueScan's Professional edition (for slides and film) costs £70. VueScan's user manual can be browsed at: **https://bit.ly/pe-mar22-vue**

If a flatbed scanner isn't for you, a number of little standalone film scanners is available, some having a built-in LCD screen. They may be ideal projects for non-computer users, and some can even copy images from 8mm and Super 8 cine film manually, frame by frame. They scan at the press of a button and files are saved to an SD card. The KODAK Mini Digital Film and Slide Scanner at £129.99 is worth a look, and there are many imported models to choose from, ranging from £60 or £70 or so. Reviews are very mixed though, so check them out carefully.

Owners of Epson's higher-end scanners might like to know that third-party film adaptors are available on eBay, Etsy and Amazon. These 3D-printed holders enable legacy formats such as 110, 126, 127, APS, Brownie or even Kodak Disc films to be digitised. As the supply of flatbed scanners becomes ever more limited, now might be the time to grab one while you can.

Onwards and upwards

In September 2021's issue I wrote about Sir Richard Branson's space launch venture, Virgin Orbit, which uses a specially adapted 747 jumbo jet to place satellites into orbit. In January, the *Cosmic Girl* 747 carrying the *LauncherOne* rocket vehicle flew from the Mojave Air and Space Port and completed a third successful mission, lofting seven satellites into orbit – the first time this orbit has been reached by launching from the West Coast, the company says. The Virgin Orbit flight had an RAF pilot at the helm and, by reaching further out over the ocean, it was able to follow a trajectory that a ground-launched vehicle would hitherto have found impossible. This shortcut to space saves time and fuel that satellites would otherwise have expended before reaching their target orbit.

Additionally, the 747 flew through weather conditions that would have grounded a conventional launchpad operation, and Virgin Orbit's mode of operation was also able to offer a very rapid turnaround for a last-minute satellite customer, they say: no need to wait for 'a full bus of passengers' before heading into space.

Trade sources say that one satellite – the ADLER-1, an acronym for 'Austrian Debris Detection Low Earth (Orbit) Reconnoiter' – was built in Glasgow, Scotland by the US satellite company Spire Global before being shipped to the US for launching. Partnering with the Austrian Space Forum and Findus Venture GmbH, the sensors on board the ADLER-1 satellite are dedicated to tracking 'micro space debris' that is orbiting the Earth.

According to infographics published by the United Nations Office for Outer Space Affairs (UNOOS) at **https://tinyurl.com/ycecrhxk**, there are 2,700 satellites sharing their orbits with 8,800 tonnes of space junk including nearly 5,000 defunct satellites and rocket stages. Although these large objects can be tracked from ground stations, there are estimated to be another 128 million small items of debris measuring 1mm to 1cm that travel like bullets and cannot be tracked from the ground. ADLER-1 is a cubesat measuring just 30 x 10 x 10cm and will use an innovative short-range CW radar that has not been tried in space until now, and a piezo-electric array will detect microparticles, also estimating the energy they impart on impact.

Virgin Orbit has now completed three successful missions and has chalked up some repeat business as well. Back home here in Britain, the next milestone of the UK National Space Strategy should be reached in Summer this year, when it's expected that Virgin Orbit will commence flights from Spaceport Cornwall (**https://spaceportcornwall.com**), the first of a series of specially adapted airstrips around the country that will enable satellites to be launched into LEO with none of the drawbacks of launchpad-based missions. The UK Government's stated intention is to capture the European small satellite launch market by 2030.

Many readers might know that Cornwall played host in 1901 to Marconi's first transatlantic radio message broadcast to Newfoundland using a spark transmitter (see **https://en.wikipedia.org/wiki/Poldhu**). Cornwall is also home to the Goonhilly Down ground station, whose fine heritage started with receiving the world's first transatlantic TV signal in 1962 via Telstar, the first communications satellite. In 1977, Goonhilly Down played a critical role in the roll-out of the Internet as we know it today – more history is at: **www.goonhilly.org/ges-heritage**. More recently, Cornwall held out the tantalising prospect of mining its own lithium, as mentioned in earlier columns – Cornwall is quite the place to be!

Autoflight's eVTOL Prosperity I flying over London (artist's impression).

Taxiing to take off

Anyone who hails a cab for trips around town may be offered another travel choice in the future: a flying shared 'taxi' is being introduced by Autoflight, the Chinese builders of electric vertical take-off and landing (eVTOL) vehicles. Autoflight produces a range of industrial eVTOL unmanned cargo drones and is now turning its attention to building a manned passenger-carrying craft called 'Prosperity I'. The proposed electric air taxi would have a range of about 250km and would carry three passengers along with (phew!) a pilot. Autoflight's European R&D hub is being established near Munich and the company is currently seeking European airworthiness certification for its aircraft, with the aim of starting passenger services in 2025.

Aldi chasing Amazon

Following hard on the heels of Amazon's Just Walk Out retail stores mentioned before in *Net Work*, the German supermarket chain Aldi, now the fifth biggest store brand in the UK, has opened its first 'cashierless store' for public testing in Greenwich, London. Called Aldi Shop&Go, it uses camera technology provided by AiFi (**https://aifi.com/**) along with an obligatory mobile app that tots up the shopping trolley automatically before charging your account on exit. Those wanting to buy a 'Challenge 25 product' like alcohol can opt for an AI facial age recognition scan provided by the secure digital ID platform Yoti (or a real human can check the buyer's age).

A ring of bogus product reviewers based in Germany has been broken up after Amazon obtained High Court injunctions against its operators. Some 20,000 reviewers in the UK alone were said to be giving five-star reviews after receiving products for which they were subsequently re-imbursed.

Finally, do check the *Net Work* blog on the *PE* website (**www.electronpublishing.com**), where the links mentioned above are ready made for you. You might also find reader feedback and more updates there too. See you next month!

The author can be reached at:
alan@epemag.net

DOWNLOAD YOUR PE/EPE BACK ISSUES

COMPLETE-YEAR collections – all 12 issues for each year!

From just £6.95 each

VOL 28 – January to December 1999 – **£6.95**
VOL 29 – January to December 2000 – **£6.95**
VOL 30 – January to December 2001 – **£6.95**
VOL 31 – January to December 2002 – **£6.95**
VOL 32 – January to December 2003 – **£6.95**
VOL 33 – January to December 2004 – **£6.95**
VOL 34 – January to December 2005 – **£6.95**
VOL 35 – January to December 2006 – **£6.95**
VOL 36 – January to December 2007 – **£6.95**
VOL 37 – January to December 2008 – **£6.95**
VOL 38 – January to December 2009 – **£8.95**
VOL 39 – January to December 2010 – **£8.95**
VOL 40 – January to December 2011 – **£8.95**
VOL 41 – January to December 2012 – **£13.95**
VOL 42 – January to December 2013 – **£14.95**
VOL 43 – January to December 2014 – **£15.95**
VOL 44 – January to December 2015 – **£15.95**
VOL 45 – January to December 2016 – **£17.95**
VOL 46 – January to December 2017 – **£20.95**
VOL 47 – January to December 2018 – **£25.95**
VOL 48 – January to December 2019 – **£25.95**
VOL 49 – January to December 2020 – **£27.95**
VOL 50 – January to December 2021 – **£29.95 – NEW!**

FIVE-YEAR collections – 60 issues!

From just £14.95 each

PE2832: Jan 99 to Dec 03 – £14.95
PE2933: Jan 00 to Dec 04 – £14.95
PE3034: Jan 01 to Dec 05 – £14.95
PE3135: Jan 02 to Dec 06 – £14.95
PE3236: Jan 03 to Dec 07 – £14.95
PE3337: Jan 04 to Dec 08 – £14.95
PE3438: Jan 05 to Dec 09 – £19.95
PE3539: Jan 06 to Dec 10 – £19.95
PE3640: Jan 07 to Dec 11 – £19.95
PE3741: Jan 08 to Dec 12 – £19.95
PE3842: Jan 09 to Dec 13 – £24.95
PE3943: Jan 10 to Dec 14 – £24.95
PE4044: Jan 11 to Dec 15 – £29.95
PE4145: Jan 12 to Dec 16 – £29.95
PE4246: Jan 13 to Dec 17 – £35.95
PE4347: Jan 14 to Dec 18 – £36.95
PE4448: Jan 15 to Dec 19 – £37.95
PE4549: Jan 16 to Dec 20 – £39.95 – NEW!

20 YEARS OF PE!

AMAZING BUNDLE OFFER
Every issue from 1999 – 2018

240 magazines for just

£55.95

PE2832: Jan 99 to Dec 03
PLUS!
PE3337: Jan 04 to Dec 08
PLUS!
PE3842: Jan 09 to Dec 13
PLUS!
PE3447: Jan 14 to Dec 18

Do you have the 15-year bundle?

Upgrade to the new 20-year bundle for just £9.95 with: PE15to20

Download your collections at: www.electronpublishing.com

Win a Microchip MPLAB ICD 4 In-Circuit Debugger

Practical Electronics is offering its readers the chance to win a Microchip MPLAB ICD 4 In-Circuit Debugger (DV164045) – and even if you don't win, receive a 15%-off voucher, plus free shipping for one of these products.

The new MPLAB ICD 4 introduces a faster processor and increased RAM to deliver up to twice the speed of ICD 3 for the in-circuit debugging of PIC microcontrollers and dsPIC digital signal controllers. The MPLAB ICD 4 also introduces a wider target voltage range and an optional 1A of power via an external power supply.

For maximum flexibility, MPLAB ICD 4 features a selectable pull-up/pull-down option to the target interface and programmable adjustment of debugging speed for greater productivity. The MPLAB ICD 4's significant improvement in speed is accomplished through a 32-bit MCU running at 300MHz. Faster processing, together with an increased buffer memory of 2MB, results in a product that is up to twice as fast as its predecessor.

Microchip's MPLAB ICD 4 is easy to use and supports many PIC microcontrollers and dsPIC digital signal controllers in Microchip's portfolio through the MPLAB X Integrated Development Environment (IDE). This simplifies the design process for customers when they choose to migrate from one PIC MCU to another to meet the needs of their application.

The MPLAB ICD 4 connects to the PC using a high-speed USB 2.0 interface and is connected to the target with a debugging connector which is also compatible with the MPLAB ICD 3 or MPLAB REAL ICE In-Circuit Emulator systems. The MPLAB ICD 4 also works with JTAG interfaces.

- Faster and more flexible real-time in-circuit debugging
- Supports many PIC microcontrollers and dsPIC DSCs
- x2 faster than ICD 3
- Reduced wait time improves debugging productivity
- Simplifies migration between PIC MCUs.

Worth $298.99 (approx £220.50) each

Free-to-enter competition

Microchip MPLAB ICD 4 In-Circuit Debugger

How to enter

For your chance to win a Microchip MPLAB ICD 4 In-Circuit Debugger or receive a 15%-off voucher, including free shipping, enter your details in the online entry form at:

https://page.microchip.com/PE-ICD.html

Closing date

The closing date for this offer is 31 March 2022.

December 2021 winner

Sam Hughes

He won a Microchip dsPIC33CH Curiosity Development Board

64-KEY ARDUINO MIDI MATRIX

Part 1
by Tim Blythman

This simple project turns an Arduino into a MIDI key matrix. These are popular with musicians for triggering samples, but commercial versions cost hundreds of pounds. Ours costs a fraction of that, and you can customise it by changing the Arduino software. It supports regular or illuminated buttons and can also be programmed to act as a MIDI pass-through, among other roles.

This project was inspired by a reader request to create something similar to the Infra-red *Remote Control Assistant* project from July 2021, but for MIDI.

MIDI is a standard that allows musical instruments and computers to communicate. Just in case you didn't know, 'MIDI' is an acronym for 'Musical Instrument Digital Interface'.

A MIDI encoder takes inputs from a musical instrument (eg, a keyboard) and converts them into MIDI format. Such a device could be connected to a computer to record playing, or a synthesiser, to turn the MIDI data back into music.

Such devices commonly utilise 8x8 switch matrices to generate up to 64 different MIDI messages; they effectively emulate a five-octave keyboard with some keys to spare. This allows you to easily interface with a synthesiser or digital audio workstation (DAW) to generate music from real-world inputs.

Working with Arduino

The Arduino community has done a lot of the work for this already, creating libraries which can generate MIDI messages both in hardware (as serial data) and also as a virtual USB MIDI device (which many DAW PC applications can read).

The basic system can be implemented with not much more than an Arduino Leonardo development board. The Leonardo is based on an ATmega32U4 microcontroller, which has a USB peripheral. Along with an Arduino library, that makes this job much easier.

To do this, the Leonardo scans columns assigned to eight of its I/O pins and checks if they have been shorted against any of eight other I/O pins, assigned to the button rows, thus giving up to 64 combinations. These 16 I/O pins are then wired to an array of tactile switches or pushbuttons which form the keys.

This simple system cannot detect more than one 'closure' at a time, so any state that is identified as having more than one button pressed is reported as 'nothing pressed'. Some, but not all, combinations of multiple keys could be identified, but we have erred on the side of keeping this simple.

To be able to detect simultaneous keypresses correctly would require a diode to be fitted to each switch (and would also make our simple device considerably more complicated).

Each key (close or release) event results in a MIDI event being sent over USB. We are using the Leonardo's hardware serial port to generate a hardware MIDI signal. This can then be fed through our *MIDI Encoder Shield*, described below, to convert it to the correct electrical format to go to a synthesiser.

Note that if all you want to do is send MIDI events to a computer over USB, you don't even need to build the Shield. But you probably will want to assemble our *Switch Matrix* PCB, also described later, since wiring up the switches manually would be a lot of work!

To help make this project more useful, we've also added a very basic synthesiser to the Leonardo. A PWM signal is produced from pin 13, approximating a sinewave at the frequency of the note being played. The waveform shape is defined in an array, so it could be changed to produce a different sound.

This sound can be heard by connecting a piezo transducer between pin 13 and GND of the Leonardo, although these devices don't have a great response to lower frequencies. Hence, our *MIDI Encoder Shield* also provides an audio amplifier which can drive a speaker (the larger the better – they're usually more efficient) for better audio quality.

While we were at it, we thought we'd also add a MIDI Input to the *MIDI Encoder Shield*. As presented here, all you can use that for is to replicate the received data directly to the MIDI Output, allowing this device to act as a basic extender. However, the hardware is set up to allow the micro on the Leonardo to receive and decode the incoming MIDI data, so with appropriate software, it could do a variety of other jobs.

The MIDI Encoder Shield

This small PCB is an Arduino ‘shield’ (aka daughterboard) which adds some useful hardware for interfacing with MIDI equipment.

The board effectively combines four different ‘modules’ which operate independently. So, if you don’t need all of the functions, you can leave off some of the parts.

These four parts are the interface to the switch matrix, an audio amplifier, a MIDI transmitter and a MIDI receiver. The circuit diagram for the whole Shield, incorporating those four sections, is shown in Fig.1.

Switch matrix

Since the switches are intended to be mounted off-board, we have just provided some convenient connection points on the PCB.

CON1 and CON2 are standard 2.54mm (0.1-inch) pitch headers, and could be fitted with pin headers or sockets. For prototyping, we recommend header sockets, as these allow jumper wires to be plugged in.

CON1A and CON2A have a 3.5mm pitch and are sized to fit smaller screw terminals such as Altronics’ P2028. This is a good way to rig up something more permanent. You could also solder wires directly to any of these pads.

Note that the pins marked with the arrows correspond to the ‘lowest’ ends of each row and column. Thus, shorting the two pins marked with arrows will give the lowest note. Shorting the two pads at the opposite ends will give the highest note.

You will probably not be able to install both of CON1 and CON1A, or CON2 and CON2A, as the headers will foul the cable entries for the screw terminals. Thus, you should choose which of the two you will fit before starting construction.

Audio amplifier

The amplifier circuit is based around IC1, an SSM2211 class-AB amplifier IC, which we previously used in the *AM/FM/SW Radio* published in December 2021. It provides a push-pull output at up to 1.5W into 4Ω, so it is a good choice for low supply voltages. A 100nF capacitor bypasses its supply rails at pins 6 and 7.

Jumper JP2 can be used to connect Arduino pin D13 to the amplifier. If you want to use another I/O pin to feed the amplifier, it can be patched into JP2. IC1 is surrounded by components to condition the input signal (including filtering out any high-frequency PWM artefacts) and to set the gain.

The 1kΩ resistor and 100nF capacitor provide low-pass filtering to remove PWM switching harmonics from the generated audio signal. This results in the 180kHz PWM frequencies being attenuated by around 40dB.

The 10µF capacitor provides AC-coupling into the amplifier, while the other 100nF capacitor provides bypassing of the internal mid-rail reference output on pin 2 of IC1. This reference rail is fed directly to pin 3, the non-inverting input, as we are supplying a single-ended signal.

The half-rail reference also biases the inverting input via the 1MΩ resistor, while the filtered audio comes into the inverting input via a 10kΩ resistor. The amplifier’s A output (at pin 5) is also fed back in to pin 4 via VR1. This is used to set the gain, and thus the resulting sound volume.

The B output at pin 8 is derived from the A output using an internal inverting stage referenced to the mid-rail voltage. Thus, the overall gain of the circuit is double the value of the feedback resistor divided by the value of the input resistor. The factor of two is due to the outputs being bridged.

VR1 can be used to linearly set the output level from zero (at 0Ω) to full-rail (at 10kΩ).

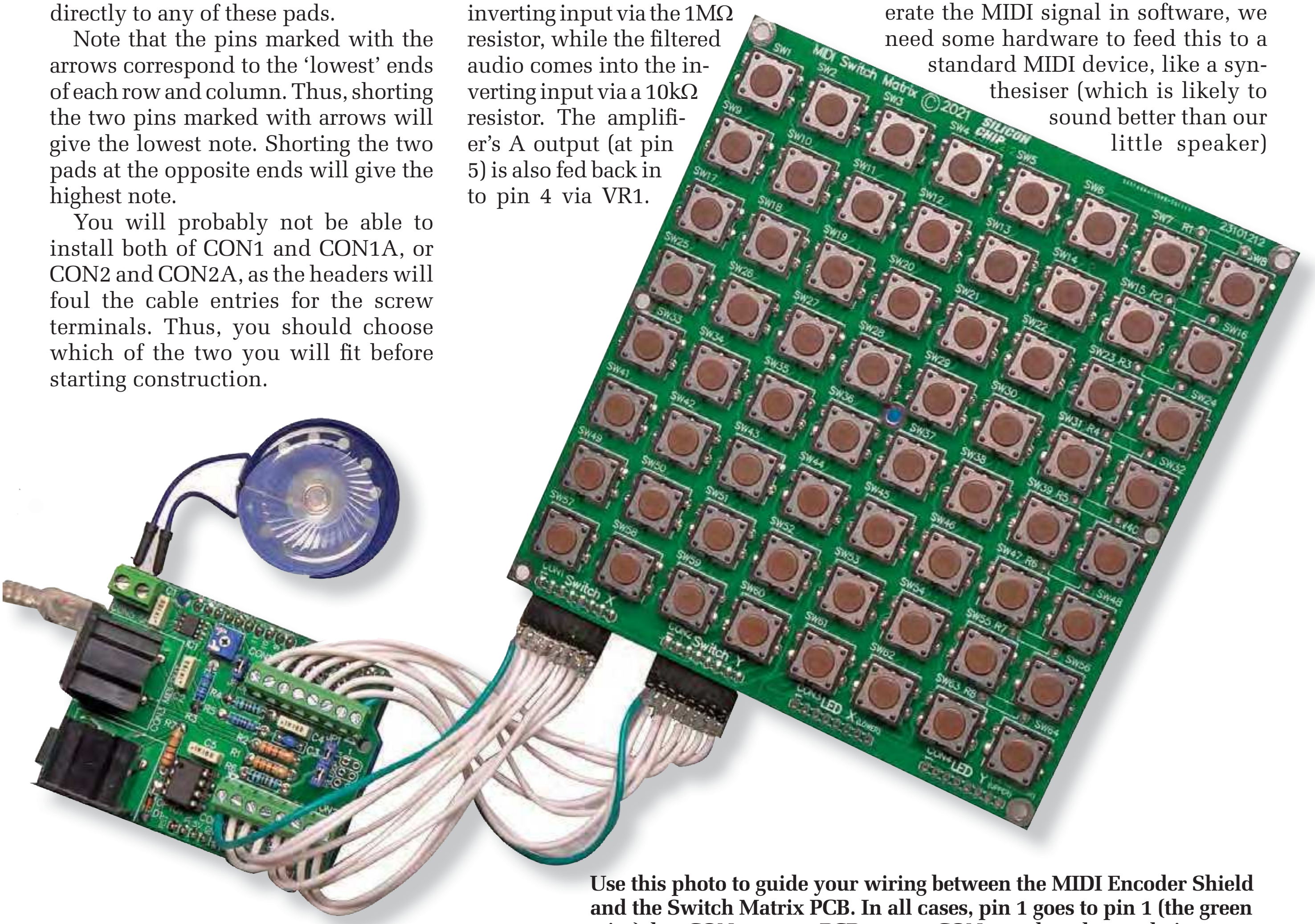
But note that this is limited by the fact that the outputs can only get within 400mV of the power rails. The push-pull outputs mean that the total maximum swing is around 8V peak-to-peak.

The complementary push-pull outputs also mean that the quiescent state has both outputs near the mid-rail level, so little (if any) direct current flows through the speaker, and thus no output coupling capacitor is needed.

The complementary outputs at pins 5 and 8 are connected to screw terminal CON5, for wiring up a 4Ω or 8Ω speaker. The SSM2211 can deliver up to 350mA, giving up to 1.5W into a 4Ω load or about 1W into an 8Ω load.

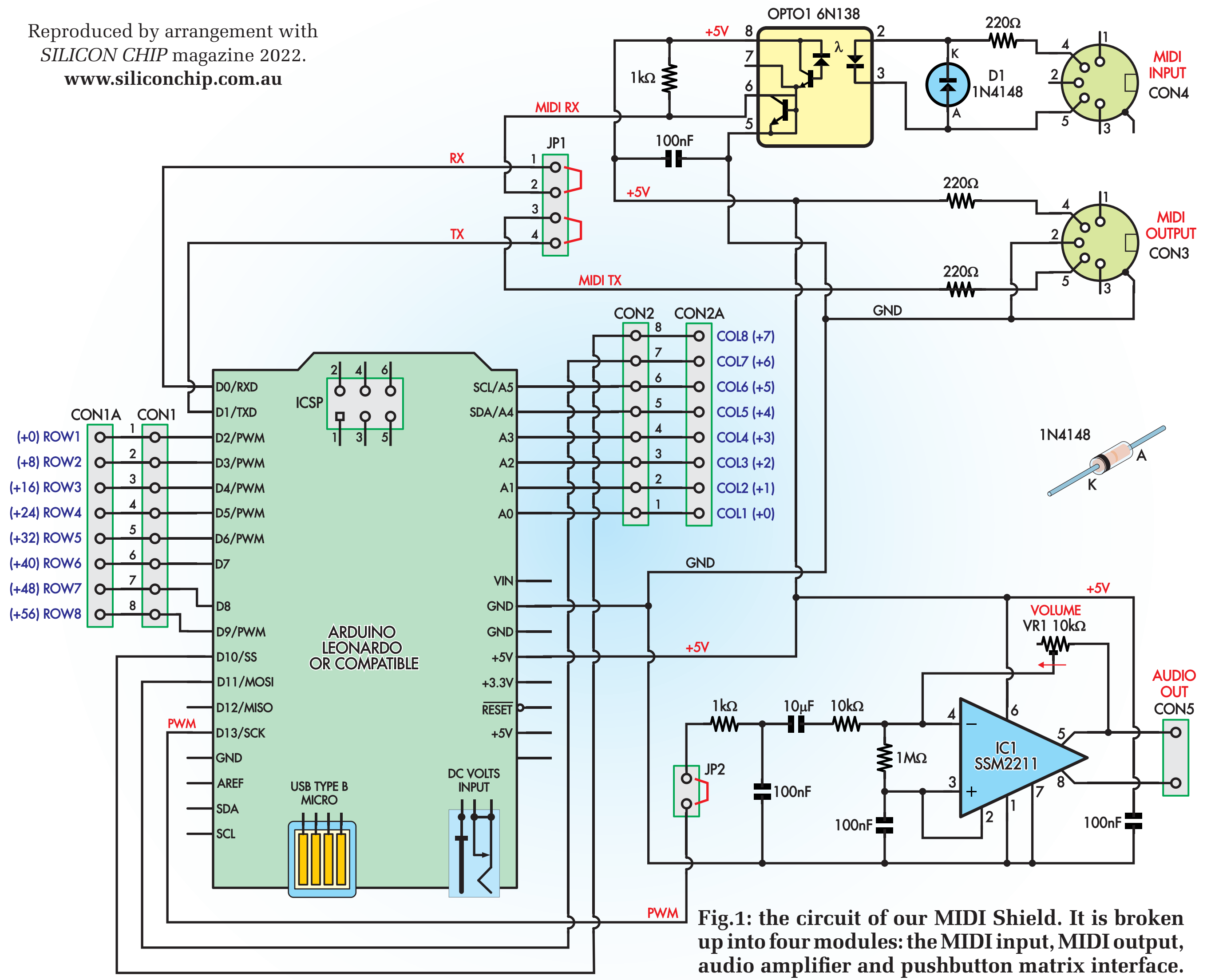
MIDI transmitter

While the Arduino Leonardo can generate the MIDI signal in software, we need some hardware to feed this to a standard MIDI device, like a synthesiser (which is likely to sound better than our little speaker)



Use this photo to guide your wiring between the MIDI Encoder Shield and the Switch Matrix PCB. In all cases, pin 1 goes to pin 1 (the green wire), but CON1 on one PCB goes to CON2 on the other and vice versa.

Reproduced by arrangement with
SILICON CHIP magazine 2022.
www.siliconchip.com.au



MIDI Shield for Arduino

Fig.1: the circuit of our MIDI Shield. It is broken up into four modules: the MIDI input, MIDI output, audio amplifier and pushbutton matrix interface. You only need to install parts corresponding to the parts you wish to use. Note that our sample software does not make use of the MIDI input (CON4).

or even a USB-MIDI converter for feeding the data to a computer.

This is quite simple, as the MIDI interface uses optoisolated connections at the receiver end.

We use two 220Ω resistors to connect the 5V supply to pin 4 of 5-pin DIN socket CON3, and the MIDI signal to pin 5. At the receiving end, we expect another 220Ω resistor and an optoisolator with a forward voltage of around 1.5V, giving a nominal 5mA current flow when our micro pin is low during data transmission.

Note that the signal from the micro must jump from pin 4 of JP1 to pin 3 to reach CON3. The signal itself is just 31,250 baud serial, easily generated by the Leonardo's UART peripheral.

JP1 allows the signal to be patched in from another pin, if you wish to use the Shield for some other MIDI application. Otherwise, you'd just leave a jumper shorting pins 3 and 4.

Note the use of the 5V rail for the return signal. Since the serial idle state is a high level, this means that no current will flow in the loop when data is not being transmitted; the same as in the disconnected state.

CON3's pin 2 and its DIN shield are connected to ground at the transmitter end only, to avoid ground loops.

MIDI receiver

As alluded to above, the MIDI receiver consists of a 220Ω resistor and an optoisolator connected between pins 4 and 5 of CON4. Diode D1 protects against a reverse voltage which another device might apply. So when pin 4 is a couple of volts higher than pin 5, the internal LED in OPTO1 is forward-biased, and thus its output transistor conducts.

The specifications for the 6N138 suggest that under adverse conditions, the propagation delay of the 6N138 could violate the timing requirements of the MIDI signal. But most MIDI designs appear to use this device without any problems.

The circuit is compatible with the 6N137 optoisolator, which, requires more current to operate, but is much faster. The nominal 5mA loop current is close to the minimum recommended for the 6N137, but should be sufficient under most conditions.

In either case, the output side of OPTO1 has power supplied at pins 8 (5V) and 5 (GND), bypassed by a 100nF capacitor. The output (pin 6) is pulled up to 5V by a 1kΩ resistor and is pulled to GND whenever the opto's LED is forward-biased.

This output signal is fed via pins 1 and 2 of JP1 to the Leonardo's UART RX pin, D1. This can also be patched into another pin if necessary.

JP1 also offers the possibility of using the *MIDI Encoder Shield* as a MIDI bridge, by placing the jumper across the middle two pins (pin 2 and pin 3). This will connect the output of OPTO1 to the transmitter at CON3, thereby passing any signal straight through.

This might not be much use on its own, but could be used in combination with a connection to the Leonardo's RX as a MIDI signal monitor or sniffer.

Switch and LED Matrix

We imagine that most people using our *MIDI Encoder Shield* will hook it up to a bunch of tactile switches in a matrix to trigger the various notes. You could do this manually, which is the cheapest option, but it would be a lot of repetitive work.

So we've designed a PCB which breaks out 64 tactile switches to a pair of eight-way headers, which can be directly connected to the headers on the *MIDI Encoder Shield* (or even straight to the Leonardo). We have even incorporated LED wiring so that you can use illuminated switches.

We've designed the *Switch Matrix* to fit the larger 12mm footprint switches, as some of these have nice big buttons that are easy on the fingers.

We also added footprints to suit small illuminated tactile switches (for example, Jaycar's SP0620/SP0622 or Altronics' S1101/S1103). These also suit the typical 6mm tactile switches, for which you might like to add keycaps (eg, 3D printed ones) for a larger key area.

If you fit illuminated switches, you can use the separate bank of eight-way headers to interface their internal LEDs. Current-limiting resistors are included for each row.

All these embellishments are optional. Since the original aim was to create a *MIDI Encoder Shield* at minimal cost, nothing is stopping you from buying a bulk lot of simple switches to populate the *Switch Matrix*.

Fig.3 shows the circuit diagram for the *Switch Matrix* with all parts fitted. The resistors are only needed if you are using illuminated switches. The LED polarity is not fixed by the PCB, but can be changed by rotating the buttons 180° on installation.

When the *Switch Matrix's* CON1 and CON2 are connected to the *MIDI Encoder Shield's* CON2 and CON1 respectively (all pin 1 to pin 1), pressing S1 will trigger the lowest note, S2 the next note and so on. You can swap or reverse the connectors to change this order.

The LEDs are similarly wired to CON3 and CON4, although there is no corresponding output on the *MIDI Encoder Shield* or Leonardo (since we've already used all the Leonardo's pins). Thus, if you want individual LED control, you'll need a separate circuit.

Later, we'll describe some sample Arduino code to light up the LEDs using simple timer-based multiplexing.

Alternatively, if you just want the LEDs to light up, you could connect all of CON3's pins to a 5V supply and CON4's pins to ground (assuming the LED cathodes are towards the top of the PCB).

Switch options

If you wish to use non-illuminated switches, then you should ensure that they suit the footprints we have used, which measure 6.5mm x 4.5mm for the smaller parts and 12.5mm x 5mm for the larger parts.

We recommend using a larger switch with a large actuator surface for ease-of-use.

The switches are installed on a 16mm pitch, so if you are using separate keycaps, make sure they are smaller than 16 x 16mm.

For illuminated parts, the footprints suit some smaller switches. One critical factor here is to check the LED polarity before fitting. This will depend on the design of your drive circuitry. For our examples, we have assumed that the LED anodes are towards the top (S1-S8) of the PCB.

Our design assumes that the pins in the corners of the switch are shorted when the button is pressed, and open the rest of the time. Since most switches have pairs of pads connected internally lengthwise, that will typically be the case.

Shield construction

Before assembling the Shield, decide which set of the four sections you will need. If you are unsure, it's probably safest to build them all.

Note that the MIDI receiver section is not used in our *MIDI Encoder Shield* software, although we would be inclined to build it anyway, as we think the Shield will be a great way to tinker with MIDI, which you might want to do in the future.

Also, you will find it is harder to fit parts later, especially after the headers have been fitted.

We will explain the construction procedure as though all parts are to be fitted, but you can omit any you don't need. Refer now to the Shield PCB overlay diagram, Fig.2, along with the same-size photo, which show which parts go where.

Start with IC1, as it is the only SMD (surface-mounted device). We chose the SOIC (small outline IC) version as the alternative is a DFN (dual flat no-lead) package, which is considerably harder to solder. We recommend that you have some solder flux paste, tweezers, a magnifier and solder wicking braid on hand when soldering this chip.

Check the orientation of the chip; pin 1 goes to the pad nearest the notch on the silkscreen. The chip itself will be marked with a bevel along one edge, which corresponds to the stripe shown on the PCB (best seen from end-on), and also with a dot near pin 1.

Apply some flux to IC1's pads on the PCB and rest the chip in place. Apply a small amount of solder to the tip of the soldering iron and touch it to one pin to tack the IC in place.

Check that the IC is flat and square with all pins within their pads. If not, adjust the IC's position with tweezers while melting the solder on the pin. Once you are happy that it is correctly placed, carefully solder the remaining pins. This can be done by applying a little more flux to the top of the pins and adding some solder to the iron's tip.

Touch the tip of the iron against where each pin meets its pad and the flux should induce a small amount of solder to run into the joint.

Don't worry about solder bridges between pins as long as the IC is correctly placed.

Once all the pins are secured, check for bridges with a magnifier. Apply more flux and rest the braid on top of the affected pins. Gently rest the iron on the braid until the solder melts and carefully pull it away from the IC once it draws up excess solder.

Clean up any excess flux using the recommended solvent. Isopropyl alcohol works well for most fluxes, although you should take care as it is flammable.

Through-hole parts

Now you can mount the resistors. Check their values with a multimeter if you are unsure of the colour codes, and match them to the values printed on the silkscreen. Next, fit the capacitors, as shown in Fig.2.

There is only one diode to mount, and it must be soldered with its cathode band aligned with the mark on the PCB silkscreen.

Trimpot VR1 will only fit in one orientation, but you might need to bend its leads slightly, after which it should snap into place.

After soldering its leads, check that it is set near its mid-point, which is a safe default.

OPTO1, like IC1, must be oriented correctly. The notch in its body should face towards the centre of the PCB, with pin 1 on the side nearest the (currently vacant) DIN sockets. We used an IC socket so that we could test out a few different optoisolators, but we recommend that you solder it directly to the PCB.

Install JP1 and JP2 next, with the jumper shunts inserted to hold the pins in place. They also provide a bit of thermal insulation if you need to manipulate the jumpers while soldering them, although this should be done with care as they can get quite hot.

Solder the headers in place, ensuring that they are flat against the PCB and straight, then move the jumpers to the default positions shown in Fig.2.

Now fit either CON1 or CON1A, and CON2 or CON2A. If you are fitting the screw terminal headers (CON1A and CON2A), ensure that these are oriented with the wire entry holes facing out of the PCB.

If you have a collection of shorter terminals, slot them together into a single block using tabs on their ends before soldering them.

Mount CON5 next. Like CON1A and CON2A, make sure that the wire entries face the edge of the PCB. Follow with the DIN sockets (CON3 and CON4). They should only fit in one way. Solder one pin and check that they are sitting correctly before soldering the remaining pins.

The final parts are the headers used to attach the Shield to the Leonardo board, which are mounted on the underside of the board.

The easiest way to manage this is to fit the headers to the Leonardo board, then slot the Shield onto the headers.

Check that the PCB is flat and if necessary, trim any long leads on the underside of the Shield that may be preventing it from sitting flat. Then solder each pin from the top side of the PCB. That completes the construction of the Shield.

Switch Matrix construction

We recommend fitting the resistors first, as they sit lower than the switches, although if you are not using illuminated switches, they are not required.

Follow by mounting the switches. If they are illuminated types, choose the orientation based on your LED wiring needs, and ensure that all LEDs face the same way. (If you don't have illuminated switches, then their orientations won't matter.)

Push each switch in place and ensure it is sitting flat before soldering. We've slightly oversized the holes to allow for some variation in parts, but the switches should still snap into place.

One good way of ensuring that they are all aligned is to insert all the buttons, then rest a flat board on top, hold onto this board and the PCB, then flip the assembly over. The flat board will align the tops of the switches.

Solder all the terminals to the PCB and trim them if they are long.

Finally, fit headers CON1, CON2, CON3 and CON4 as needed. We used socket strips on our prototypes, as we had a handful of pre-made eight-way cables that we could run directly to the headers on the *MIDI Encoder Shield* PCB. We suggest that you figure out how you will be mounting the board (see below) before soldering these, and test-fit the headers/cables, as that might affect what connectors you need.

You could solder ribbon cable straight to the pads on the *Switch Matrix* PCB and then to the *MIDI Shield* PCB; that is the cheapest way to connect the two boards.

But headers make the wiring removable, which can be handy.

If you are soldering wires to the board, then you could run them to the underside of the PCB if you will be mounting it on spacers.

If you need those wires to be pluggable, you could mount right-angle pin headers on the underside.

Note on our photos that the wire from pin 1 of CON1 on the *MIDI Encoder Shield* PCB goes to pin 1 of CON2 on the *Switch Matrix* PCB, and pin 1 of CON2 on the *MIDI Encoder Shield* PCB goes to pin 1 of CON1 on the *Switch Matrix* PCB.

Mounting the Switch Matrix

Despite the small space available, we've squeezed seven M3 mounting holes into the design.

Some of these might not be usable depending on the switches you have chosen, although an M3 screw should still fit in the central hole, even with 12mm switches fitted.

The PCB material is strong, but repeated flexing from enthusiastic keypresses could fatigue it, so we recommend mounting it to something solid, like a piece of plywood.

Use some short spacers or a stack of washers to provide clearance for the component leads under the PCB.

Wiring up switches manually

If you really want to do it this way, you can. Wire up the switches in rows and columns like in our circuit (see Fig.3).

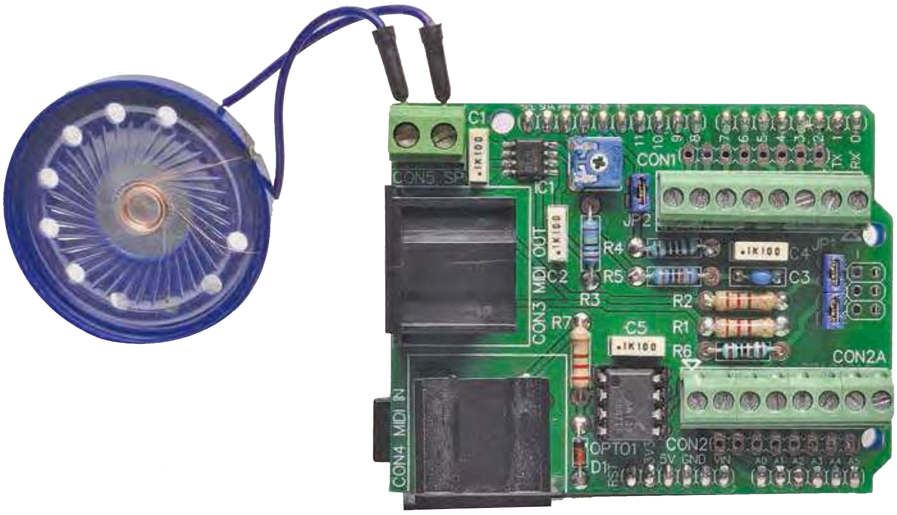
If you have built the *MIDI Encoder Shield*, connect the rows and columns to CON1 and CON2 respectively, with the ends going to the lowest-numbered switch at pin 1 in each case.

If you're using a bare Leonardo to pass MIDI messages to a computer via USB, you can instead use Fig.1 as a guide for the wiring, as it shows how rows and columns connect to the Arduino pins.

You can connect a piezo buzzer between pin D13 and the adjacent GND for sound output, and raw MIDI data is available at pin D1 (TX), referenced to one of the GND pins.

Software

If you don't already have the Arduino IDE (integrated development environment) installed on your computer, download it from **www.arduino.cc/en/software** (it's free and available for Windows, Mac and Linux).



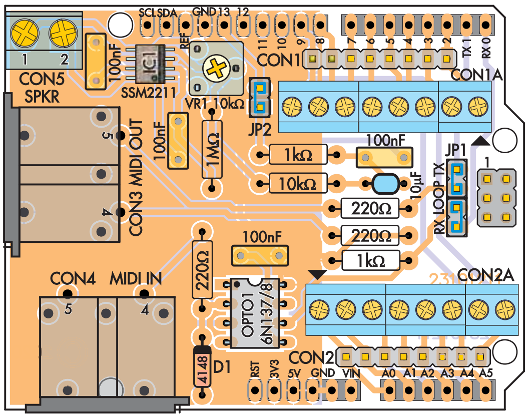
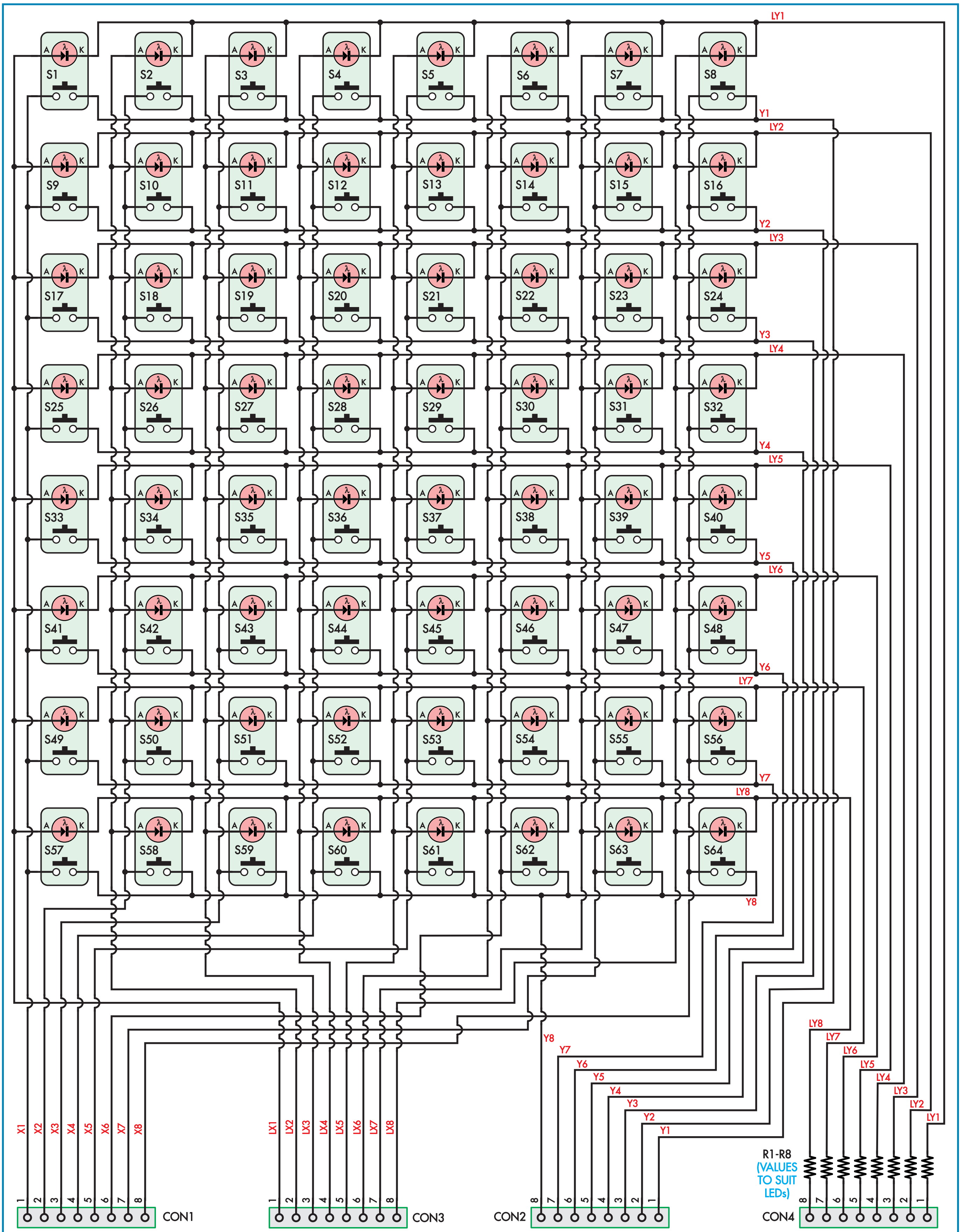


Fig.2: use this overlay diagram and the photo above as a guide when assembling the MIDI Shield PCB. Apart from IC1 (which is the amplifier for the speaker), all parts are common through-hole types. While IC1 is an SMD, it can be soldered without any special tools, although we recommend using tweezers and a magnifier. Solder it first so that you aren't restricted by nearby parts.



MIDI Shield Switch Matrix

Fig.3: there isn't much to the Switch Matrix circuit. Each pin of each connector goes to either a row or column of contacts on the switches or LEDs.

If you already have the IDE, check that you are using a recent version (at least 1.8.x). While developing this project we were using version 1.8.12.

Launch the IDE and open the Library Manager (Sketch -> Include Libraries -> Manage Libraries) and search to 'TimerOne'. This library is used to provide regular timer interrupts to produce the audio waveform. Install it now, if you don't already have it on your system.

Parts list – MIDI Shield

1 double-sided PCB coded 23101211, 69 x 54mm, available from the PE PCB Service
1 Arduino Leonardo module
2 6-way pin headers (part of the Arduino shield headers)
1 8-way pin header (part of the Arduino shield headers)
1 10-way pin header (part of the Arduino shield headers)

Switch Matrix interface parts
2 8-way pin headers or sockets (CON1,CON2) ***OR***
2 8-way 3.5mm screw terminals (CON1A,CON2A) [eg, 8 x Altronics P2028]
wiring to switch matrix

Audio amplifier parts
1 4-8Ω 1W loudspeaker
1 2-way 5/5.08mm-pitch screw terminal (CON5)
1 2-pin header and jumper shunt (JP2)
1 SSM2211 audio amplifier, SOIC-8 (IC1)
3 100nF 63V MKT capacitors
1 10µF through-hole ceramic capacitor (ideally 5.08mm lead pitch)
1 1MΩ 1% 1/4W metal film resistor
1 10kΩ 1% 1/4W metal film resistor
1 1kΩ 1% 1/4W metal film resistor
1 10kΩ mini horizontal trimpot (VR1)

MIDI output parts
1 5-pin, 180° DIN socket, right-angle PCB mount (CON3)
[eg Jaycar PS0350, Altronics P1188B]
1 4-pin header and jumper shunt (JP1)
2 220Ω 1% 1/4W metal film resistor

MIDI input parts
1 5-pin, 180° DIN socket, right-angle PCB mount (CON4)
[eg Jaycar PS0350, Altronics P1188B)
1 4-pin header and jumper shunt (JP1)
1 6N138 optoisolator, DIP-8 (OPTO1)
1 1N4148 small signal diode (D1)
1 100nF 63V MKT capacitor
1 1kΩ 1% 1/4W metal film resistor
2 220Ω 1% 1/4W metal film resistors

Parts list – 8x8 Switch Matrix
1 double-sided PCB coded 23101212, 131 x 140mm, available from the PE PCB Service
64 tactile pushbutton switches* (S1-S64) – see text
2 8-way pin headers (CON1,CON2)
16 female-female DuPont jumper leads (to connect CON1 and CON2 to the MIDI Shield)
M3 mounting screws and spacers to suit your application (optional)
** we used Diptronics DTS-21N-V (non-illuminated, from Mouser) and C&K ILSTA250 30 (illuminated, from Digi-Key). Jaycar SP0620/SP0622 and Altronics S1101/ S1103 are also suitable alternatives.*

Extra parts for illuminated switches
2 8-way pin headers (CON3,CON4)
16 female-female DuPont jumper leads (for CON3 and CON4)
8 1/4W axial resistors to suit LEDs

The second library we need is called MIDIUSB and can be found by searching for 'MIDIUSB' in the Library Manager. The final library is called 'MIDI Library'. Several different libraries are found in a search for 'midi', so you should see our screenshot (Screen1) to verify that you have found the correct library.

We've included the zipped versions of all three libraries in our download package; installed via the Sketch -> Include Library -> Add Zip Library menu option. (Download from the April 2022 page of the *PE* website.)

MIDI Library is set up to use the hardware serial port on the Leonardo's pins D0 and D1, with the MIDI data being produced at the TX pin, D1. We tested this with an Arduino synthesiser sketch, and it worked as expected.

Once all the libraries are installed, open the 'MIDI_ENCODER' sketch. Select the serial port of the Leonardo and upload the sketch to the Leonardo.

In our sketch, the SAMPLE_RATE define is set to an integer number of microseconds between interrupts (this is done to minimise any rounding errors).

This is followed by the sinewave data, as 256 unsigned integer bytes (0-255).

The matrix pin definitions follow this. The rows each contribute a multiple of +8 to the key number, while the columns contribute +1. The key count thus spans zero to 63, and is offset by the START_NOTE value, which we've set to 28. That means that the *MIDI Encoder Shield* will produce notes from E1 (about 41Hz) up to G6 (1568Hz), centred near middle C (262Hz).

The range is limited to 64 notes by the matrix size, but changing the start note changes where that range spans.

The notes[] array sets the frequencies that are produced on pin D13. You might want to tweak these if they don't sound right or you prefer a different scale.

Some parameters associated with the library follow. These set the channel and velocity that are used in the data that is sent. The defaults should work with most software, although some programs might map channel 0 (in the Arduino code) to channel 1.

Testing the Shield

At this stage, you should have built the Shield, plugged it into the already programmed Arduino Leonardo and attached the *Switch Matrix* (or whatever switches you will be using).

Plug the Leonardo into a USB port, launch the Arduino IDE (if it isn't already running), make sure the correct COM port is still selected, then open the Arduino Serial Monitor.

Start pressing buttons in the *Switch Matrix*, one at a time. You should see the Serial Monitor report that S1 causes UP/DOWN actions on MIDI note 0. This is because we've started the switch numbering at 1, but the MIDI notes begin at 0.

If you check that the four corner keys are correct (the switch number is one more than the note number), then the remaining keys are probably correct. If you find that you get incorrect notes, try flipping the connections end for end at CON1 or CON2 on the *Switch Matrix* PCB, or swapping CON1 for CON2.

Test all the keys; if you see any single keypresses not being detected, check the PCB for bad solder joins on the corresponding switch.

Usage

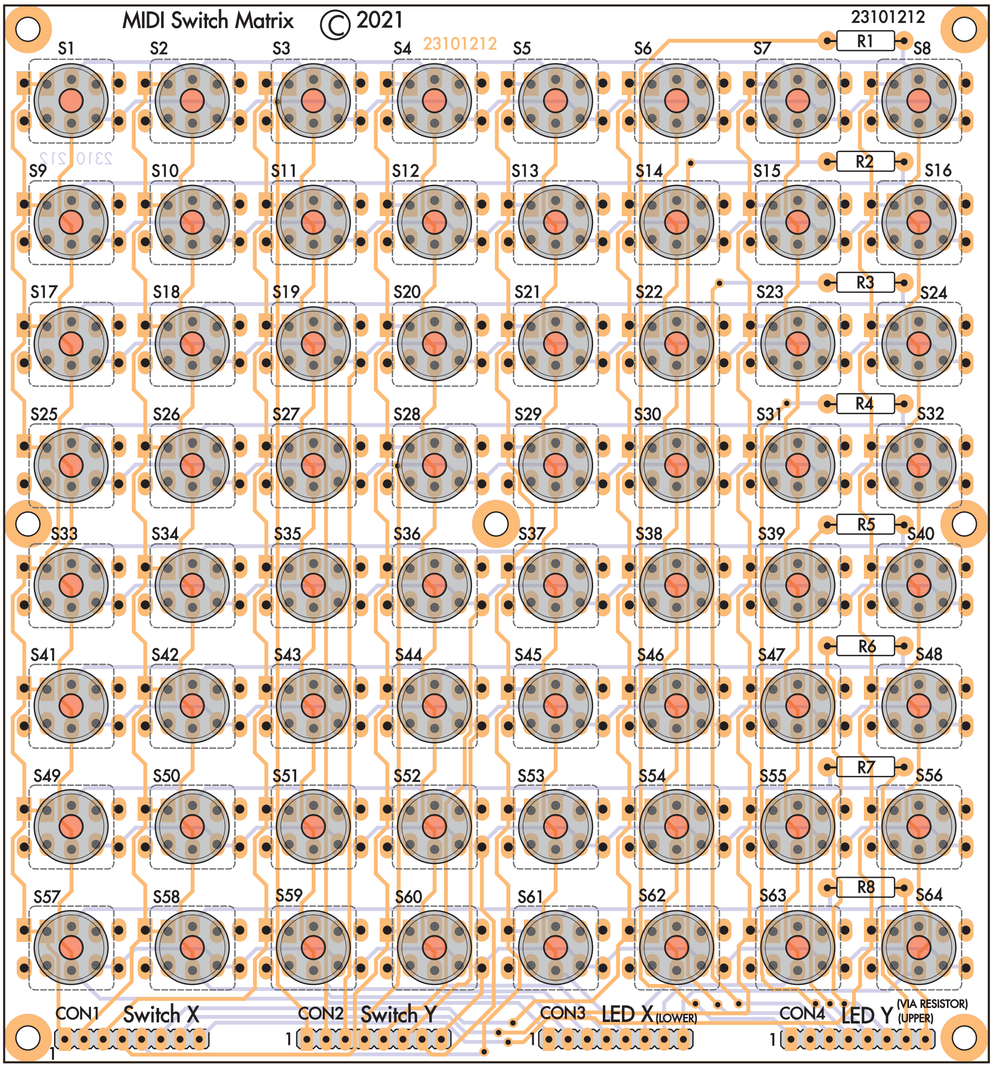
After the sketch is uploaded, the Leonardo appears as a native USB-MIDI device to a computer.

As well as the audio and USB and hardware MIDI outputs, the Leonardo also prints information to the serial monitor (accessible from the Leonardo's native USB-serial port) about which note is being played. This can be handy for testing.

We used a program called Muse-Score (**https://musescore.org/en**) to test that the computer was correctly receiving MIDI data.

It automatically detected that a MIDI device was present and played

Fig.4: overlay diagram for the Switch Matrix PCB. This is shown with 12mm large non-illuminated tactile switches in place; they fit to the four pads just outside the switch footprint. The next set of four pads are for smaller 6mm switches, while the innermost two pads are for the LEDs of illuminated switches. Most illuminated switches are reversible, so that the LEDs can be installed with either polarity.



synthesised piano sounds, although it can also transcribe to played notes among other features.

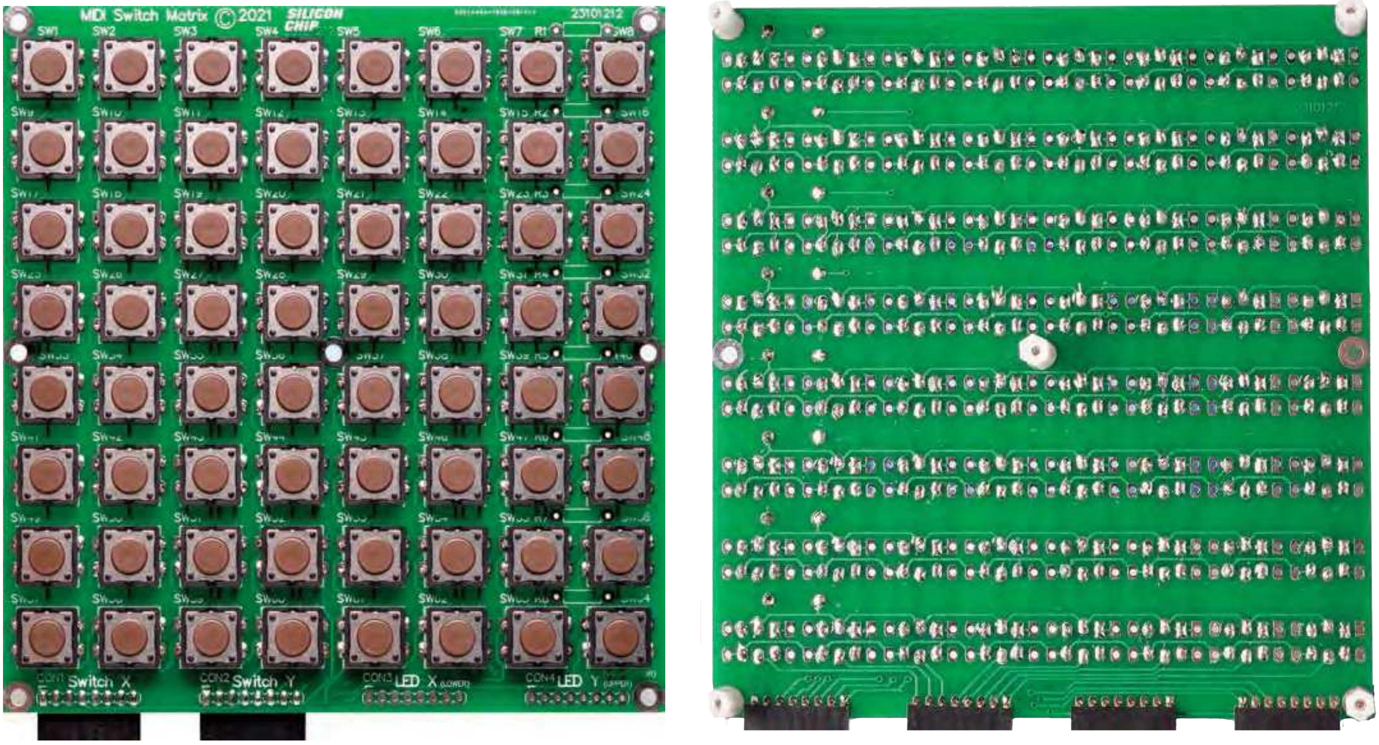
Without a computer, you will have to connect something to the audio output (on pin 13) or the hardware MIDI data (on pin 1).

As we noted, we tested another Arduino sketch which worked as a synthesiser (this sketch expects MIDI data on the Arduino's serial port; typically pin 0). This might be a better option if you would like to get better sounds without much expense.

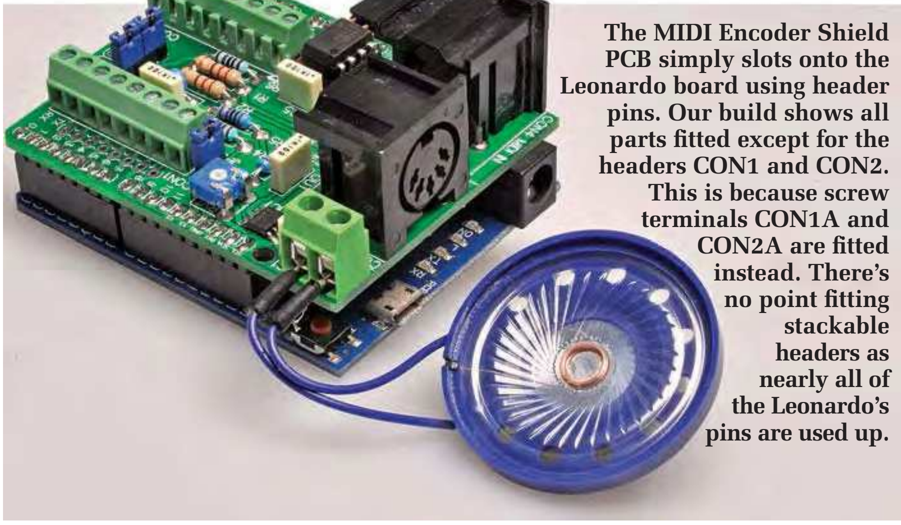
Note that it is possible to trigger sounds from the Shield using simple jumper wires. Anything that connects one of the row wires to one of the column wires will trigger a sound.

Using something like a cheap membrane matrix keypad could be a simple way of adding an input device, especially if you want to create percussion sounds. Four of the 4x4 matrixes could be connected to give the full complement of 64 inputs. Just make sure to wire up each matrix to a different combination of row and column wires.

Of course, since we've included the



(Left) The non-illuminated version of the Switch Matrix. (Right) The rear of the illuminated version; note the extra leads, tapped standoffs and CON3 and CON4 fitted at the bottom. These are shown about 70% life size. The actual PCB size is 131 x 140mm.



The MIDI Encoder Shield PCB simply slots onto the Leonardo board using header pins. Our build shows all parts fitted except for the headers CON1 and CON2. This is because screw terminals CON1A and CON2A are fitted instead. There's no point fitting stackable headers as nearly all of the Leonardo's pins are used up.

Arduino source code, you can use it as a starting point for creating your own MIDI-based project.

LED test sketch

We have created a test Arduino sketch to light the LEDs if you have fitted illuminated switches. We're assuming that you've fitted them with the anodes to the top, as we did on our prototype. We also assume that you've got the Arduino IDE installed, including the libraries for the *MIDI Encoder Shield*. We only need the TimerOne library for this sketch.

Open the 'MATRIX_LED_DRIVER' sketch and upload it to the Leonardo. Connect CON3 of the *Switch Matrix* PCB to CON2 on the *MIDI Encoder Shield*. Then connect CON4 of the Switch Matrix PCB to CON1 on the *MIDI Encoder Shield*.

You should see a diagonal row of LEDs light up. You can change the starting state with the LED[] array, and manipulate this in the loop() function to animate.

We've also created a self animating version 'MATRIX_LED_DRIVER_GOL', which implements a simple *Conway's Game of Life* simulation on the 8x8 matrix. The array is loaded with a pair of 'gliders', which move as long the *Switch Matrix* is powered. You can find out more at: **https://bit.ly/pe-ape22-cgol**

Using it

If you've built the amplifier section, now would be a good time to wire up a speaker. Generally, a short length of twin-core cable is all you need to connect it up, and most speakers have tabs that suit soldering or quick-connect spade lugs. The other ends of the wires can then be screwed into CON5. The polarity doesn't matter much as the output at CON5 is AC.

With the *MIDI Encoder* sketch uploaded to your Leonardo, you should be able to get a tone from the speaker by connecting any of CON1's pads to any of CON2's pads (eg, by pressing a button on the attached key matrix). However, we found that our small speaker was not able to render the lower notes too loudly.

If the audio is distorted, reduce the volume by turning VR1 anti-clockwise. The mid-point should be audible for practically all speakers, so if you can't hear anything, check your construction before increasing the volume.

The *MIDI Encoder Shield* output socket (CON3) can be connected to the MIDI input port of another device, such as an electronic piano or DAW (digital audio workstation).

Similarly, the *MIDI Encoder Shield* input connection (CON4) can be driven from another device's MIDI output port.

Note that CON4 does not do anything with our default software, as it is not programmed to have any function.

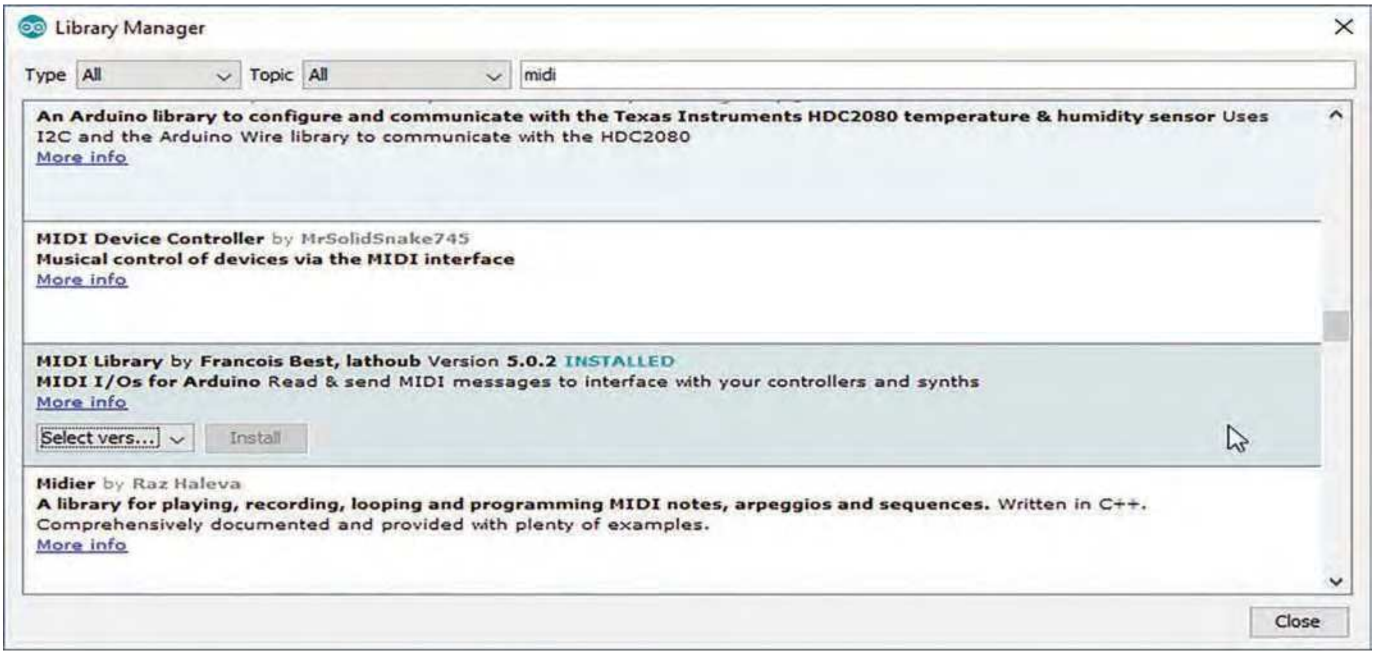
Conclusion

While originally intended as a simple bit of hardware to make better use of the *MIDI Encoder Shield* software, we think that this Shield will be handy for anyone who wants to dabble in custom MIDI hardware.

It could, for example, be used as a MIDI synthesiser by using the hardware *MIDI Encoder Shield* input (CON4) or USB MIDI input (in software) to receive MIDI messages and turn them into sounds from the speaker.

In a follow-up article, we will show how to control illuminated pushbuttons from our MIDI sketch. This requires some extra hardware, as the Leonardo doesn't have enough pins free to do this by itself.

We will also describe how to connect this device to a smartphone or tablet running Android, and install a MIDI synthesiser app which can then be controlled using the *Key Matrix*.



midi

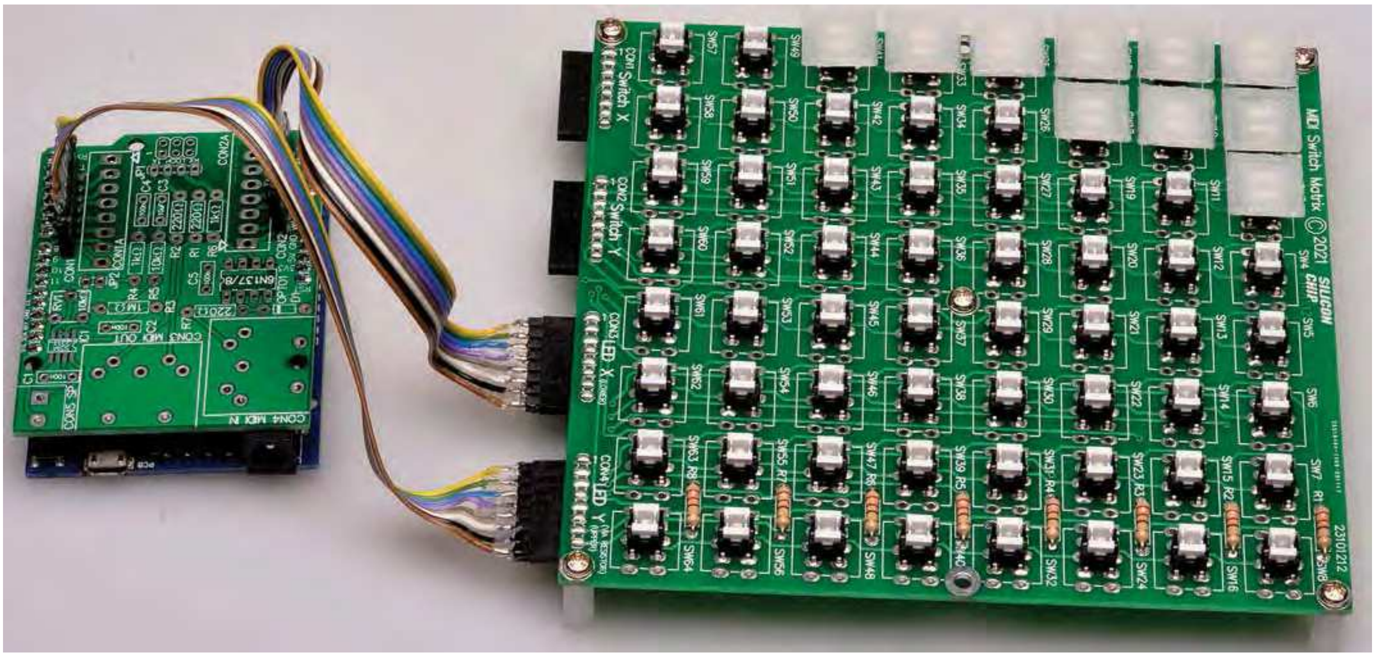
An Arduino library to configure and communicate with the Texas Instruments HDC2080 temperature & humidity sensor Uses I2C and the Arduino Wire library to communicate with the HDC2080
More info

MIDI Device Controller by MrSolidSnake745
Musical control of devices via the MIDI interface
More info

MIDI Library by **Francois Best, lathoub** Version **5.0.2** **INSTALLED**
MIDI I/Os for Arduino Read & send MIDI messages to interface with your controllers and synths
More info

Midier by Raz Haleva
A library for playing, recording, looping and programming MIDI notes, arpeggios and sequences. Written in C++. Comprehensively documented and provided with plenty of examples.
More info

Screen1: the Arduino Library Manager will give a lot of results for a 'midi' search, so use the one highlighted here or use the zip version which can be downloaded from the April 2022 page of the PE website.



This is the deluxe version of the Switch Matrix PCB, with illuminated switches, although you'll have to provide your own keycaps – a good use for your 3D printer! (3D Keycap files can be downloaded from the April 2022 page of the PE website.). We've fitted it with standoffs to prevent the pointy leads from damaging the surface it's on or, conversely, shorting out on any conductive surface. Note the Shield PCB at left fitted with only headers to allow it to be used as a USB MIDI device only.

We also intend for this article to contain some more detailed information on the MIDI protocol, for those who wish to expand upon our software, or are just interested to learn how it works.

Note that the *Switch Matrix* presented here could be useful in many other contexts; it doesn't have to be used for MIDI. It can serve as a general-purpose switch array with up to eight rows and eight columns; you don't even need to populate all the switches. For example, you could wire up the row and column headers to an Arduino Mega board (or similar) and use it as a general keyboard, to type in letters and numbers etc (with suitably labelled keycaps).

And as the Mega has many more pins than the Leonardo, it could also easily drive the LED matrix to light up keys as they are typed, or show which keys are valid inputs, for example.

MATRIX

☎ 01422 252 380
www.matrixtsl.com

We are looking for an experienced electrical design engineer in West Yorkshire.

Who are we?

At Matrix, we develop industrial-standard training equipment for education providers around the world. Our solutions cover Science, Engineering and Technology disciplines from Robotics, Electrical engineering and Automotive learning to Manufacturing and Mechanical Engineering.

The role:

We are looking for an electrical design engineer of products and solutions, for educational institutions. They will:

- Understand market trends and subject matter and have a track record of explaining technology through technical writing
- Have broad experience in digital and analogue electronic system design
- Have experience of programming in C/C++ or equivalent and a good understanding of electronic system design software such as Proteus
- Be capable of managing projects and teams to deliver solutions for teaching in the global engineering education market
- Be educated to degree level in engineering
- Be able to take a hands-on approach to problem solving, both in day-to-day production and the development of new products.

If you think this could be you, please email a covering letter and CV to **recruitment@matrixtsl.com**

Digital FX Unit

Part 1

by John Clarke

Make like a pro muso with this Digital FX (Effects) Unit. It will produce unique sounds when connected to a variety of instruments… like an electric guitar, bass, violin or cello, even the output of a microphone preamp or within the effects loop of an amplifier or mixer.

It's very common for musicians to add 'effects' to the sound of their musical instruments. These are not only used to add depth, ambience and tonal qualities, but also to personalise the sound. Effects can be subtle or extreme, and can be tailored to produce a unique sound.

Purely analogue audio circuitry can be used for effects units, such as in the *Overdrive and Distortion Pedal* from March 2021. But for complex effects, digital signal processing (DSP) is more convenient and flexible.

Our *Digital FX Unit* uses a DSP (digital signal processing) IC (integrated circuit) that is designated the 'SPN1001 FV-1' (or 'FV-1' for short). This is preprogrammed with eight effects, and while one of these is a test function, the remaining seven provide flange, chorus and tremolo as well as pitch shift and reverb effects.

A further eight extra effects are stored within an external EEPROM that connects to the FV-1. These effects have been chosen by us. However, you can change the stored effects patches.

The FV-1 has been available for many years, and has been used in many commercially available effects units. In fact, the FV-1 has something of a cult following among digital effects enthusiasts. This has led to the production of numerous freely available effects patches and software to enable the writing of your own unique effects.

For our *Digital FX Pedal*, the preprogrammed EEPROM is filled with eight effects that add to the seven usable effects preset within the FV-1. These individual effects are selected using a rotary control knob, while the parameters of each effect are adjusted using up to three rotary controls.

Many effects have already been created for the FV-1 IC, and these are free to use. These effects include chorus, echo, flange, phase shift, vibrato, limiter, wah, various reverberation effects, distortion, octave shifts and a ring modulator. For information on what some of these effects are and how they are achieved, see: **https://bit.ly/pe-apr22-dfx1**

We will explain some of the basic effects at the end of this article.

There is also an assembler and a graphical software package to help you write your own effects if you feel inclined to experiment. The software can then be assembled and programmed into the EEPROM.

This requires an EEPROM programmer; we will have more details on where to get effects patches, how to store them in EEPROM and how to use the assembler and graphical software later.

Presentation

Our *Digital FX Pedal* is designed for live music use, and so is housed in a rugged diecast aluminium case. On the top, it has a footswitch, eight rotary controls plus indicator LEDs.

The signal inputs and outputs are two 6.35mm (1/4-inch) jack sockets at the rear, along with a DC barrel socket for power. The unit can also be powered via an internal 9V battery. Its power is automatically switched on when a jack plug is inserted into the output socket.

Features

- 15 different effects including chorus, echo, flange, vibrato, wah, reverb and distortion
- Each effect has up to three adjustable parameters
- Provision to experiment by adding new effects
- Rugged enclosure, suitable for stage use
- Power supply reversed-polarity protection
- High input impedance to suit piezo pickups
- Low power consumption
- Battery or DC plugpack power
- True bypass switch
- No signal phase inversion

Fifteen different effects are available, with the option to change eight of the effects to your liking. You can choose them from a list of many freely available effects, or create them yourself using freely available tools.

Operating principle

The block diagram in Fig.1 shows the signal flow of the *Digital FX Pedal*. The original signal is applied to CON1, and this is connected to the bypass switch (S2a).

When not bypassed, this signal goes to the high-input-impedance buffer (IC1a) and is then filtered with a 19kHz low-pass filter. This prevents unwanted artifacts in the subsequent DSP stage, by removing RF and ultrasonic signals.

The signal from the filter is fed to two separate level controls, VR1 and VR2. VR2 sets the level applied to the signal mixer (more on this later), while VR1 sets the signal level applied to the SPIN FV-1. VR1 is required so that the level can be set below the clipping level for the FV-1 input. The clip LED lights up to indicate signal limiting when the level is too high.

The SPIN FV-1 contains a stereo analogue-to-digital converter (ADC), a DSP core and a stereo digital-to-analogue converter (DAC) to produce the output signals. All processing is done using 24-bit digital audio samples. For more information, see: **https://bit.ly/pe-apr22-dfx2**

Note that while the FV-1 can process stereo signals, the *Digital FX Pedal* is a mono device, so we are only using a single channel.

There are two versions of the pedal, where the effects selection is made using either a rotary switch (S3) or potentiometer (VR8) and associated components – more about this later. The effect parameters are adjusted using potentiometers VR5, VR6 and VR7. The FV-1 also has inputs for the crystal oscillator and EEPROM serial connections.

After processing within the FV-1, the output signal goes through a 19kHz low-pass filter, to remove high-frequency noise (DAC step artefacts) and then to the effects level control, VR3. Both the effects signal and the original (or 'dry')

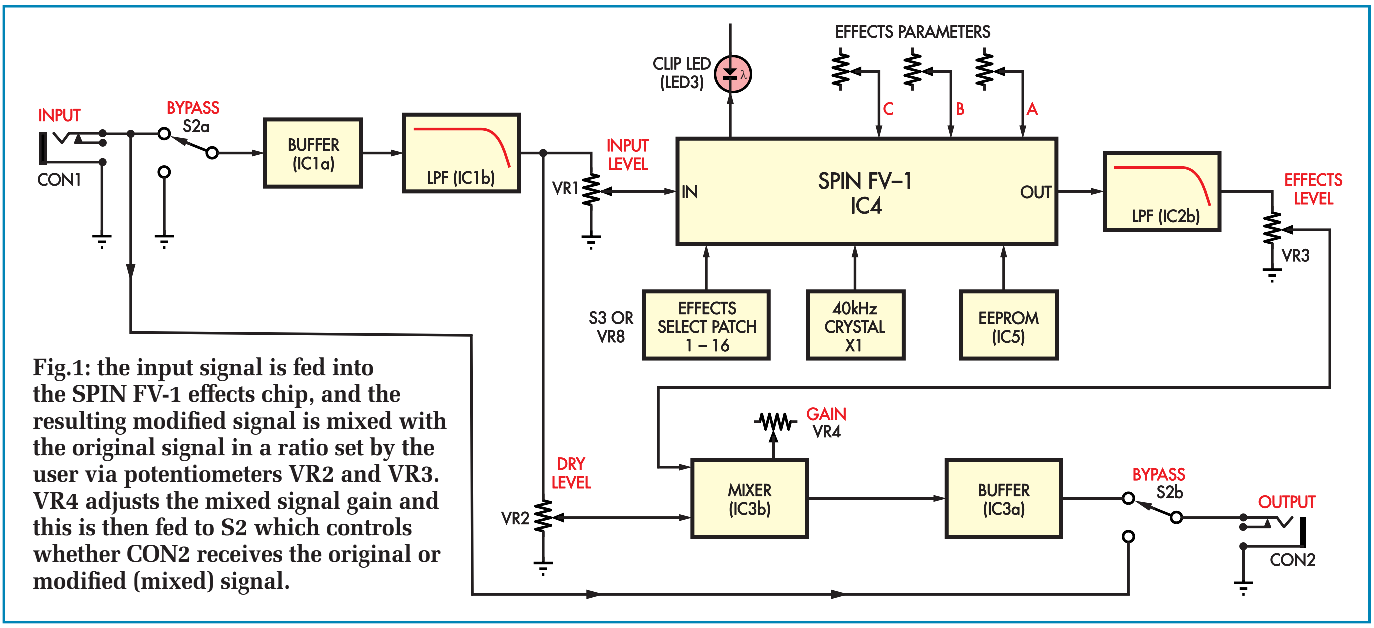
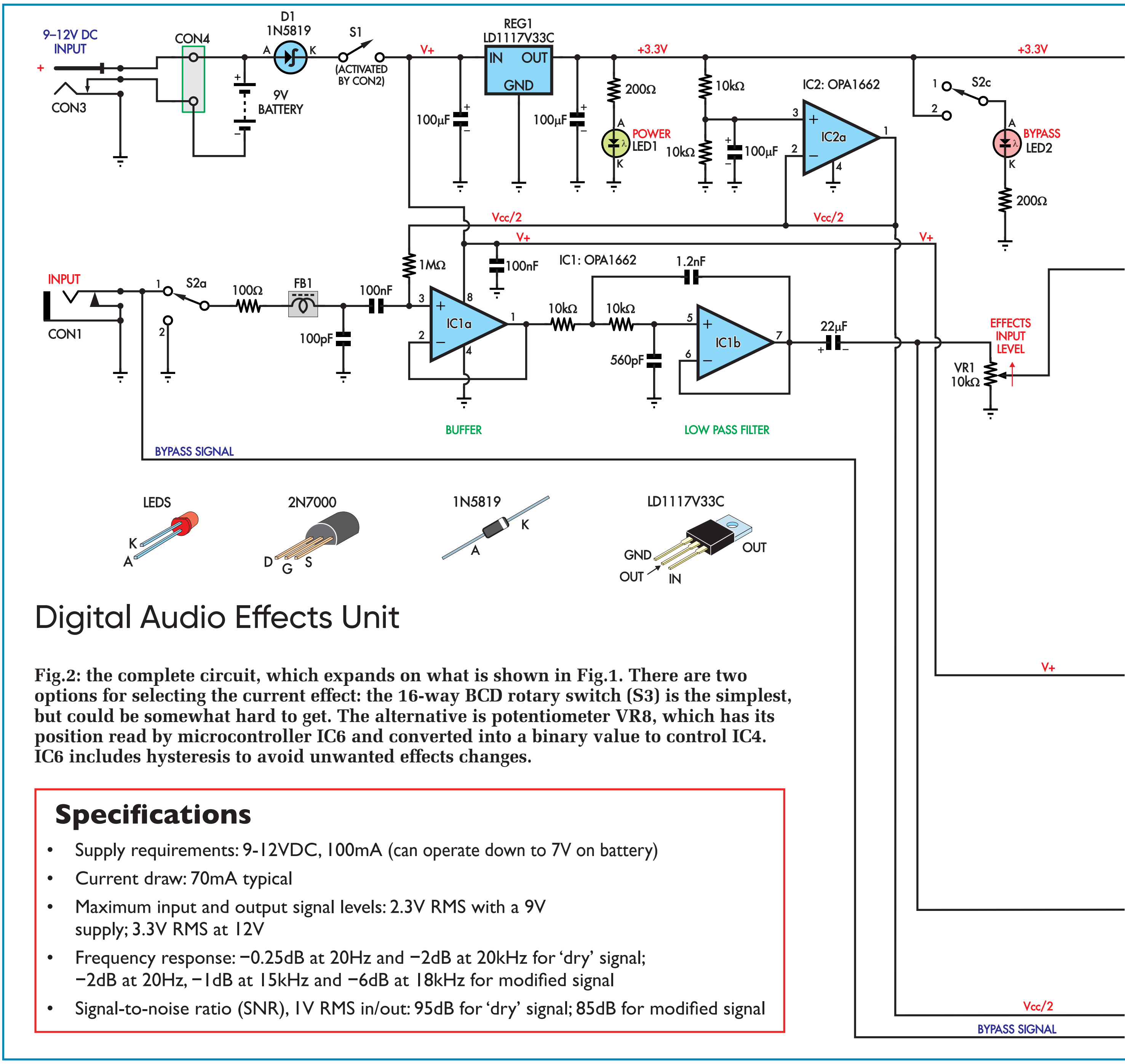


Fig.1: the input signal is fed into the SPIN FV-1 effects chip, and the resulting modified signal is mixed with the original signal in a ratio set by the user via potentiometers VR2 and VR3. VR4 adjusts the mixed signal gain and this is then fed to S2 which controls whether CON2 receives the original or modified (mixed) signal.



Digital Audio Effects Unit

Fig.2: the complete circuit, which expands on what is shown in Fig.1. There are two options for selecting the current effect: the 16-way BCD rotary switch (S3) is the simplest, but could be somewhat hard to get. The alternative is potentiometer VR8, which has its position read by microcontroller IC6 and converted into a binary value to control IC4. IC6 includes hysteresis to avoid unwanted effects changes.

Specifications

- Supply requirements: 9-12VDC, 100mA (can operate down to 7V on battery)
- Current draw: 70mA typical
- Maximum input and output signal levels: 2.3V RMS with a 9V supply; 3.3V RMS at 12V
- Frequency response: −0.25dB at 20Hz and −2dB at 20kHz for 'dry' signal; −2dB at 20Hz, −1dB at 15kHz and −6dB at 18kHz for modified signal
- Signal-to-noise ratio (SNR), 1V RMS in/out: 95dB for 'dry' signal; 85dB for modified signal

signal from VR2 are combined in the inverting mixer stage, comprising IC3a and IC3b. The mixing allows adjustable portions of the dry and effects signal to be blended to provide the desired result.

The mixer can also provide a signal gain of up to five times, adjusted with potentiometer VR4. The resulting signal is then applied to the bypass switch, S2b. This selects between the original signal from CON1 and the signal with effects, with the selected signal going to the CON2 output.

How the effects work

While it is difficult to show many of the various effects available, the 'octaver' effect can be easily demonstrated. This is where the dry signal is mixed with a signal shifted up or down by one octave. These are harmonically related, at half the frequency and double the frequency, respectively.

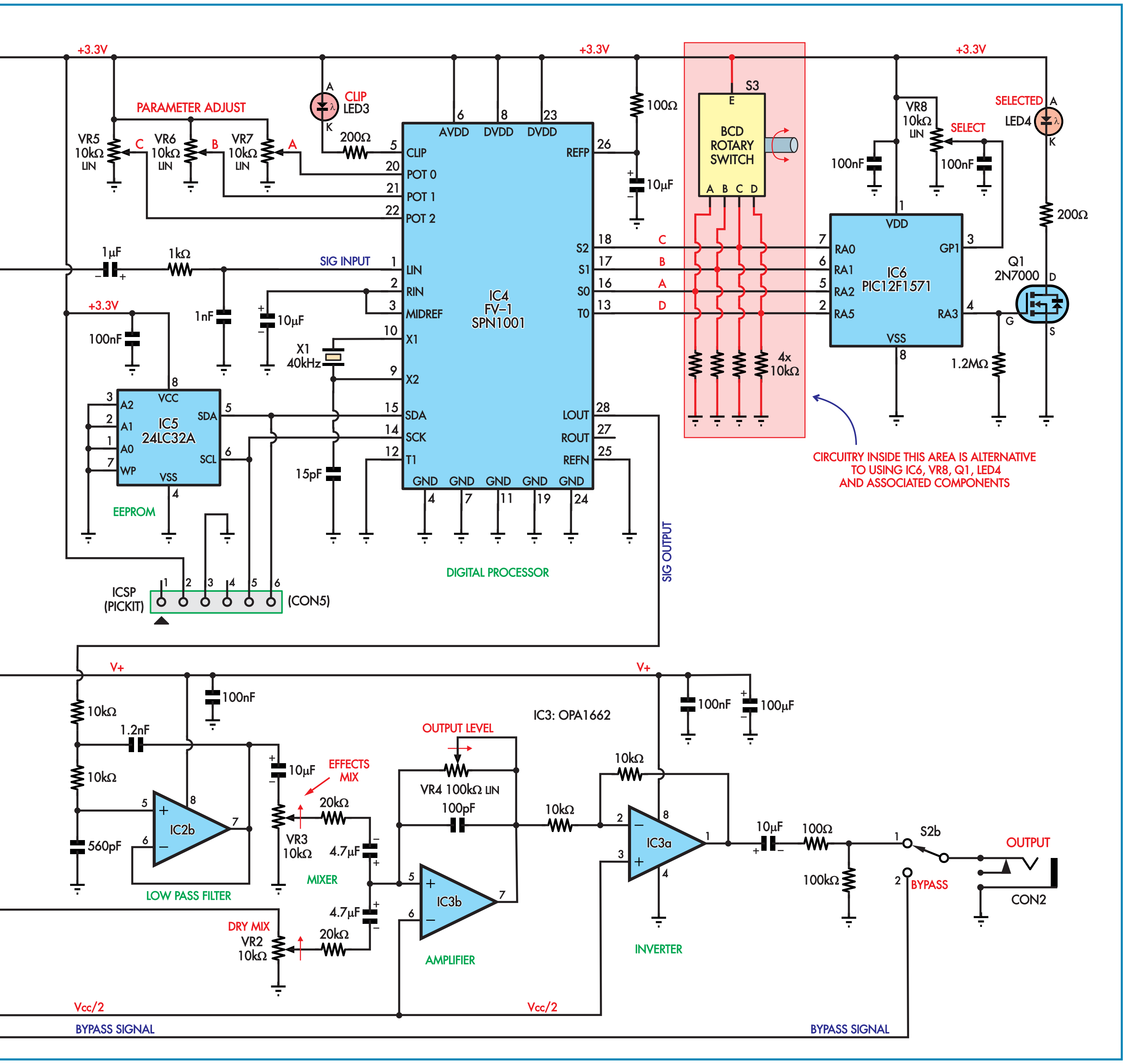
In Scope1 (overleaf), the top yellow trace (channel 1) shows the dry signal and the lower white trace (Ref A) the up-octave signal, produced by doubling the frequency. The middle blue trace (channel 2) is the down-octave signal, at one half the frequency. The up /down octave signals can be mixed along with the dry signal to produce the desired effect.

Circuit description

The full circuit for the *Digital FX Pedal* is shown in Fig.2. The input signal from CON1 passes through a 100Ω stopper resistor and ferrite bead FB1. In conjunction with the 100pF capacitor, these block RF signals from entering the circuit and causing radio-frequency detection and reception. The 100pF capacitor also provides a suitable load for piezo string pickups.

The signal is AC-coupled to pin 3 of IC1a, and is biased to half-supply ($V_{CC}/2$ or about 1.65V) via a 1MΩ resistor. This keeps the input impedance reasonably high at 1MΩ, suitable for a piezo pickup. IC1a is connected as a unity-gain buffer that can drive the following low-pass filter stage.

The $V_{CC}/2$ voltage is derived using two 10kΩ resistors connected in series across the supply and is bypassed with a 100µF capacitor to remove supply noise, then buffered by unity-gain amplifier IC2a.



Note that all the op amps in the circuit have very low noise and distortion figures of 0.00006% at 1kHz at a gain of 1, with a 3V RMS signal level. Therefore, the op amps do not contribute any audible distortion to the signal.

The low-pass filter is a Sallen-Key two-pole 19kHz Butterworth type that rolls off at 40dB per decade (12dB per octave). It is included along with further passive filtering to remove any high-frequency signal components above 20kHz. This prevents signal aliasing due to digital sampling at 40kHz. Without the filter, strange audible artifacts could be generated by the ADC.

Following this filter, the signal is AC-coupled to the level potentiometer, VR1. This sets the signal level applied to input pin 1 of IC4, the FV-1.

IC4 provides an internal half-supply DC bias for this pin, hence the AC coupling. The 1kΩ resistor and 1nF capacitor after the AC-coupling capacitor attenuate any remaining high-frequency noise.

The signal fed to IC4 must be lower than about 1V RMS to avoid clipping. Clipping occurs when the signal goes beyond the 0-3.3V supply range of IC4. The clip indicator output (pin 5) goes low and drives LED3 if this happens. VR1 should be adjusted so that this LED does not light.

IC4 includes a crystal oscillator amplifier. The typical circuit for the FV-1 depicts the crystal as a 32,768Hz watch type. This is recommended mainly because it is commonly available, but the high-frequency audio response suffers if one is used.

Instead, we use a 40kHz crystal, extending the processor's frequency response from around 16kHz (when using the watch crystal) to just under 20kHz. (Recall the Nyquist theorem, which states that the highest frequency that an ADC can handle is at half the sampling rate.)

Effects IC4 requires several supply bypass capacitors. These are 100nF for the analogue and digital 3.3V supply pins, while the half-supply bypass at the MID pin (pin 3) is 10µF. As mentioned above, the mid supply is about 1.65V.

IC4 also requires positive and negative reference voltages for the ADC at pins 25 and 26. Pin 25 is tied directly to GND (0V), while pin 26 connects to the +3.3V supply via a 100Ω resistor and with a 10µF filter capacitor, to keep supply noise out of the signal path.

Effects parameters are varied with potentiometers VR5, VR6 and VR7. These are connected across the 3.3V supply and can provide voltages of 0-3.3V to the POT2, POT1 and POT0 inputs of IC4. Each pot's function depends on the selected effect.

Effects are selected by the state of IC4's digital inputs S0, S1 and S2 (pins 16, 17 and 18) and the voltage level at the T0 input, pin 13. When the T0 input is low, the effects selected by the S0, S1 and S2 inputs are those that are preprogrammed within IC4.

If all the S0, S1 and S2 inputs are low, the first effect is selected. Further effects are chosen with different levels at S0, S1 and S2. S0 is the least-significant bit, and S2 is the most-significant bit of a binary value. The three inputs provide for eight possible selections (2^3).

The effects stored on the EEPROM (IC5) are selected when the T0 input is high (3.3V). Eight further selections are then available using the S0-S2 inputs. IC4 connects to the EEPROM via an I^{2}C serial bus using two pins, the serial clock, SCL and serial data SDA. These connections are also brought out to the ICSP header for in-circuit programming of the EEPROM memory chip (if required).

The EEPROM's supply is bypassed by a 100nF capacitor. The EEPROM is a 32kbit (32,768 bit) memory arranged as 4096 x 8bits (ie, 4k bytes). Effects patches stored within the EEPROM are placed in memory blocks of 512 x 8bit. There are eight 512 x 8bit memory blocks in the full 4k x 8bit memory.

Output signal handling

The effects signal from the left channel output of IC4 (pin 28) is fed to a low-pass filter comprising IC2b, two 10kΩ

Parts list – Digital FX Unit

1 double-sided PCB coded 01102212, 86 x 112mm* [available from thr PE PCB Service]
3 panel labels (one front, two sides – see opposite)
1 diecast aluminium enclosure 119 x 94 x 34mm [Jaycar HB5067]
2 6.35mm PCB-mount jack sockets (CON1,CON2) [Jaycar PS0195]
1 PC-mount barrel socket, 2.1mm or 2.5mm ID (CON3) [Jaycar PS0520, Altronics P0621A]
1 2-pin vertical polarised header, 2.54mm spacing (CON4) [Jaycar HM3412, Altronics P5492]
1 2-pin polarised plug (CON4) [Jaycar HM3402, Altronics P5472 and 2 x P5470A pins]
1 6-way pin header with 2.54mm spacings (CON5)
1 C&K ZMA03A150L30PC microswitch or equivalent (S1) [eg Jaycar SM1036]
1 3PDT footswitch (S2) [Jaycar SP0766, Altronics S1155]
1 Lorlin BCK1001 16-way 4-bit binary-coded switch* (S3) [RS Components 655-3162]
6 B10kΩ linear pots (VR1-VR3,VR5-VR7) [Altronics R1946]
1 B100kΩ linear pot (VR4) [Altronics R1948]
7 11.5mm-diameter 18 tooth spline (6mm) knobs (see text for special requirements) [Altronics H6560, RS Components 299-4783]
1 13mm-diameter D-shaft knob* [Jaycar HK7717]
1 ferrite RF suppression bead 4mm OD x 5mm (FB1) [Altronics L5250A, Jaycar LF1250]
1 40kHz crystal (X1) [Citizen CFV-20640000AZFB or similar; RS components 1849668]
1 9V battery clip lead (optional)
1 9V battery (optional)
1 PC stake (GND point)
1 solder lug (for grounding the enclosure)
4 M4 x 10mm Nylon screws or stick-on rubber feet (see text)
2 9mm-long M3 tapped nylon standoffs (support for PCB rear)
3 M3 x 6mm panhead machine screws (for solder lug and standoffs)
1 M3 nut and star washer (for solder lug)
1 50mm length of medium-duty green hookup wire
1 6.3mm mono jack plug or jack-to-jack lead (for testing)

Semiconductors

3 OPA1662AID dual op amps, SOIC-8 (IC1-IC3) [RS Components 825-8424]
1 SPN1001-FV1 digital FX processor, wide SOIC-28 (IC4) [**www.profusionplc.com/parts/spn1001-fv1**]
1 24LC32A-I/SN EEPROM, SOIC-8, programmed with 0110221A.hex (IC5)
1 1N5819 1A schottky diode (D1)
1 LD1117V33C 3.3V low-dropout regulator (REG1) [RS Components 6869767]
1 3mm high-intensity green LEDs (LED1)
2 3mm high-intensity red LEDs (LED2, LED3)

Capacitors

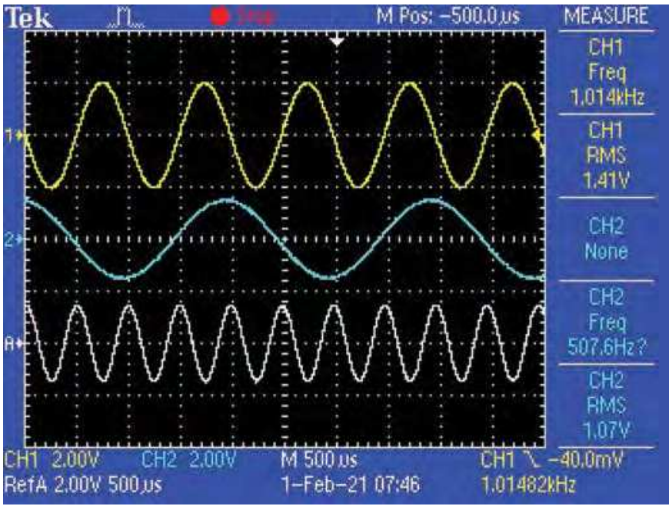
4 100µF 16V PC electrolytic
1 22µF 16V PC electrolytic
4 10µF 16V PC electrolytic
2 4.7µF 16V PC electrolytic
1 1µF 16V PC electrolytic
5 100nF MKT polyester
2 1.2nF MKT polyester
1 1nF MKT polyester
2 560pF ceramic
2 100pF NP0/C0G ceramic
1 15pF NP0/C0G ceramic

Resistors (all 1/4W, 1% metal film axial)

1	1MΩ	*(Code brown black black yellow brown)*
1	100kΩ	*(Code brown black black orange brown)*
2	20kΩ	*(Code red black black red brown)*
12	10kΩ*	*(Code brown black black red brown)*
1	1kΩ	*(Code brown black black brown brown)*
3	200Ω	*(Code red black black black brown)*
3	100Ω	*(Code brown black black red brown)*

Parts for version using a potentiometer for effects selection (delete items marked * above)

1 double-sided, plated-through PCB coded 01102211, measuring 86 x 112mm, available from the PE PCB Service
1 B10kΩ linear potentiometer (VR8) [Altronics R1946]
1 11.5mm-diameter 18 tooth spline (6mm) knob (see text for special requirements) [Altronics H6560, RS Components 299-4783]
1 9mm-long M3 tapped Nylon standoff (support for rear of PCB)
1 M3 x 6mm panhead machine screw (for standoff)
1 PIC12F1571-I/SN 8-bit microcontroller programmed with 0110221A.hex, SOIC-8 (IC6)
1 2N7000 N-channel small-signal MOSFET (Q1)
1 3mm high-intensity red LED (LED4)
2 100nF MKT polyester capacitors
1 1.2MΩ 1/4W 5% carbon axial resistor
8 10kΩ 1/4W 1% metal film axial resistors
1 200Ω 1/4W 1% metal film axial resistor



Scope1: the signal being fed into the device is shown at the top, in yellow. Below are the outputs of the 'octaver' effect, set for one octave lower (blue) or higher (white). These effects signals can be mixed into the original to create richer harmonics and different sounds.

resistors plus 560pF and 1.2nF capacitors. This is another Sallen-Key two-pole 19kHz Butterworth low-pass filter. It is included to remove high-frequency DAC switching artifacts from the signal. The output signal from IC2b is applied to the VR3 effects level potentiometer.

The signals from the wipers of VR3 and the dry signal potentiometer (VR2) are combined in the inverting mixer stage based on IC3b. The mixer gain is adjusted using VR4, with a maximum gain of negative five times when VR4 is at its maximum resistance of 100kΩ. The following inverter stage, built around IC3a, re-inverts the signal so that the output signal is in-phase with the input.

The output of IC3a is fed via a DC blocking capacitor and stopper resistor to the bypass switch, S2b. When in position 1, this signal goes to the CON2 output socket.

When bypass is selected (with S2 in position 2), the input signal at CON1 bypasses the effects circuitry, connecting directly to the output at CON2 via the S2b terminals. The S2a terminals tie the input for IC1a to ground. This prevents noise from being picked up and amplified by IC1a in bypass mode.

The remaining switch pole (S2c) controls indicator LED2. This bypass LED is lit when the signal is bypassed; the 200Ω resistor from cathode to ground limits the LED current to around 6.5mA.

Two effects selection options

So, effectively, a binary value of 0000-1111 (0-15 decimal) is required to select one of the 16 possible effects. This value controls the states of the S0-S2 and T0 inputs of IC4. Our circuit provides two ways to make this selection.

The simple way is to use a BCD (binary-coded decimal) switch, which has 16 positions and four outputs that provide the required binary states. However, 4-bit BCD switches can be difficult to obtain, so we offer the alternative option of using a potentiometer instead.

Circuit Version 1 uses a potentiometer (VR8) and a microcontroller (IC6) to convert the voltage from the potentiometer's wiper to a BC (binary-coded) value. VR8 connects across the 3.3V supply and can provide 0-3.3V to the pin 3 analogue input of microcontroller IC6. This voltage is internally converted to a digital value.

The micro's digital outputs at RA2, RA1, RA0 and RA5 then generate the required binary (0V or 3.3V) values to feed to the S0, S1, S2 and T0 inputs of IC4 respectively. The resulting binary value varies smoothly from 0-15 decimal as VR8 is rotated from fully anticlockwise to fully clockwise.

Common effect descriptions

Reverb	Several delayed versions of the original sound are mixed back with the original dry sound, to simulate sound in a room or area where there are sound reflections (a complex form of echo). The ideal reverb period or delay setting depends on the type of sound; for music, it depends on the music's tempo. As a general rule, longer reverb times are for slow tempo music, while shorter reverb times are suited to faster tempo tunes. Different reverb programs will have their own tonal qualities due to differences in the reverb time of high or low frequencies and differences in the reverb sound's overall frequency response. Be careful not to apply too much reverb, particularly in the high frequencies, as this will result in an unnatural sound (unless that's what you want!). Start with reverb level all the way down, then gradually bring the reverb mix up until you can just hear the difference. Any more than this will give an unrealistic sound.
Phasing, chorus, and flanging (modulation effects)	All of these effects have a portion of the audio signal delayed and then mixed back with the dry signal. The amount of delay is modulated by a low-frequency oscillator (LFO). The delay is quite short compared to the reverb effect. For phasing effects, the delay is less than the period of the signal. This phase difference between the modulated and direct signals causes cancellation at some frequencies and reinforcement at others. It produces a comb-filter-like effect, where some frequencies are amplified, and others are attenuated across the audio band. It causes a 'shimmering' type of sound. Phasing is the subtlest of all these effects, producing a gentle shimmer that can add life to a wide range of sources, but without being too obtrusive. For chorus and flanging, the signal is delayed by a longer period, up to several milliseconds, with the delay time modulated by an LFO. This also produces a comb-filter effect and a pitch-shift effect after mixing with the dry signal, giving a harmonically rich 'swirling' or 'swishing' sound. Chorus and flanging effects mainly differ in the amount of delay time and feedback used. Flanging uses longer delay times compared to chorus, and chorus generally uses a more complex delay structure. Chorus is most often used to 'thicken' the sound of an instrument, while flanging is usually used to produce other 'whirling' sounds.
Pitch and octave shifts	These effects involve altering the frequency of the signal. Pitch varies the frequency by a variable amount, while the octave shift changes the frequency by a factor of 0.5 for octave-down and 2.0 for octave-up. Mixing the octave-shifted signals with the dry signal produces various effects, including making a single instrument sounding fuller, or sounding as though there are multiple instruments.

Hysteresis is included to avoid the binary value flicking between two adjacent values near each voltage threshold. This requires you to rotate the selection pot a little clockwise further than the threshold to select the next higher BC value output, and a little further anticlockwise from the

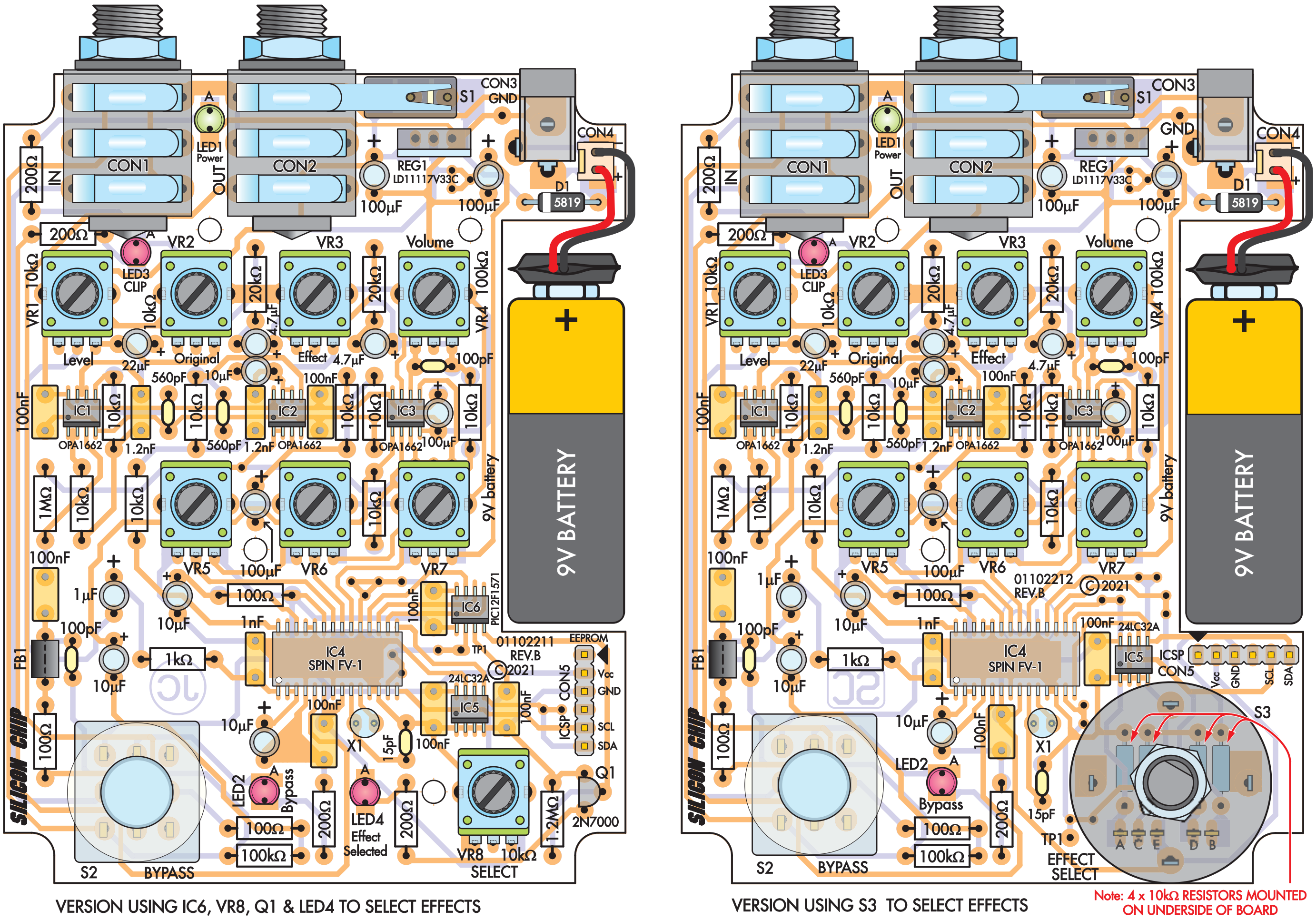


Fig.4: two versions of the project have been designed, as described in the text. Each uses a slightly different PCB, so make sure you order the appropriate board. Note that in the switched version, four resistors are mounted on the PCB underside.

threshold to select the next lower BC value. A change from one effects selection to another is indicated using LED4. The LED flashes off and then on again as the pot is rotated, to indicate a change in the binary value.

Typically, an 8-pin PIC microcontroller does not have sufficient pins to handle the analogue sensing, 4-bit binary output and the indicator LED drive. We solve this by using the master clear ($\overline{\text{MCLR}}$) input at pin 4, and task it as a general-purpose input to drive the LED.

It might seem that an input cannot be used as an output, but this input includes the option of a selectable pull-up current. While many of the 8-pin microcontrollers include an internal pull-up when the $\overline{\text{MCLR}}$ input is set to operate as a master clear input, there are not many microcontrollers that also allow the pull-up to be switched on or off when this pin is used purely as an input. However, the PIC12F1571 does have that capability.

To be used as an output, the internal pull-up current is enabled, so the input will be pulled high near to the 3.3V supply. The input will go low without the pull-up when there is an external pull-down resistor. The pull-down resistor must be sufficiently high in resistance to allow the internal pull-up current to pull the input high enough to switch on the following stage.

Using a 1.2MΩ resistor as the pull-down resistance, the minimum pull-up current for that input at 25µA is sufficiently high to swamp the pull-down current from the resistor. Thus, this pin will be quite close to 3.3V with the pull-up engaged.

We use a 2N7000 N-channel MOSFET (Q1) to convert the high-impedance drive from this 'output' to a low-impedance drive for the indicator LED. It then drives LED4 via the 3.3V supply and 200Ω current-limiting resistor when its gate is high.

The second version of the circuit (Version 2) simply uses a 4-bit BCD rotary switch (S3) to select the effect. This requires 10kΩ pull-down resistors at the A, B, C and D switch pins. The common E pin connects to 3.3V, and so pulls a combination of the A-D pins high, depending on the switch's rotation.

Power supply

The circuit is powered when microswitch S1 is activated by inserting the output jack plug into CON2. The plug physically raises the socket's ground connection, lifting the microswitch actuator and activating the switch.

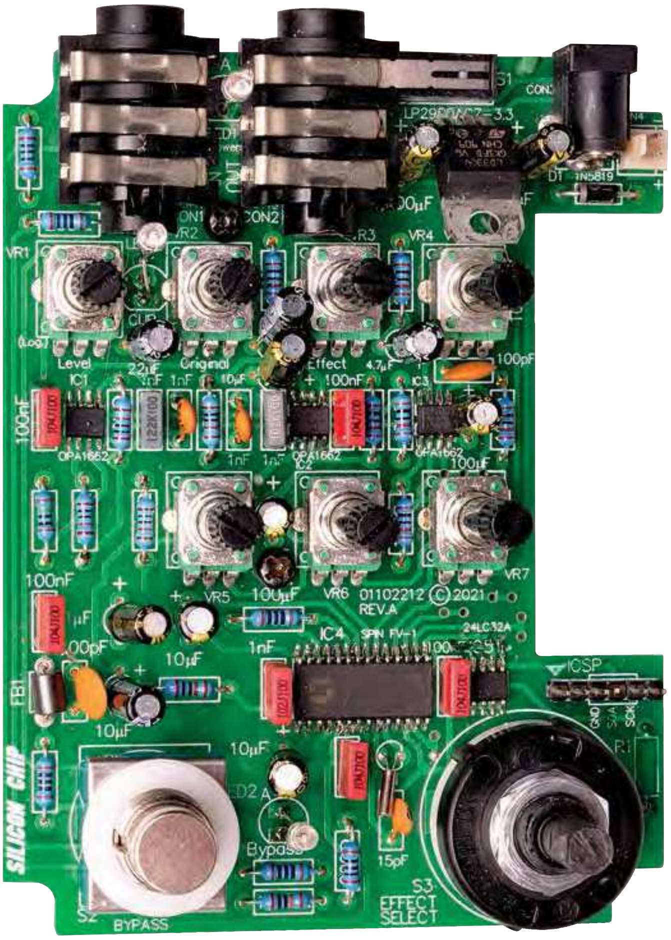
While many effects pedals are switched on when a jack plug is inserted, it is usually done by a switch internal to the socket.

We are not using a socket that has isolated switching mainly because they are not commonly available. These also have the disadvantage of stressing the PCB connections each time a jack plug is inserted, especially if the jack is moved at an angle to the socket. This eventually causes the solder joints to harden and break.

While the sockets we use also solder directly to the PCB, the body is secured to the case at the socket entry as well. That keeps the socket fixed in place against the enclosure side, minimising movement of the solder joints.

Power is automatically selected between 9V battery or DC supply. When there is no DC power plug inserted, the DC socket (CON3) will supply battery power via its normally-closed switch connecting the negative of the battery to ground. When a power plug is inserted, power is via the DC input and the battery negative is disconnected.

Power switch S1 connects power to the rest of the circuit whether via the battery or an external source, while diode D1 provides reverse-polarity protection.



Same-size photo of the switched version, the version at right opposite. The cutout is for a 9V battery, as shown.

REG1 is a low-dropout 3.3V regulator, which supplies IC4, IC5 and IC6 (if used). The input and output pins of REG1 are bypassed with 100µF capacitors. Its output drives the power LED (LED1) via a 200Ω resistor.

Construction

The *Digital FX Pedal* is built using a double-sided, plated-through-hole PCB measuring 86 x 112mm. The version using the BCD switch is coded 01102212, while the version using potentiometer VR8 is coded 01102211. Both are available form the *PE PCB Service*. Either way, it is housed in a diecast enclosure measuring 119 x 94 x 34mm.

Figs.3 and 4 are the two PCB overlay diagrams for the different versions. Refer to the appropriate diagram during construction to see which parts go where.

Begin by fitting the surface-mount parts, IC1-IC5 (and possibly IC6), on the top side of the PCB. These are not difficult to solder using a fine-tipped soldering iron. Good close up vision is necessary, so you might need to use a magnifying lens or glasses. If you're using the version with potentiometer VR8, also mount IC6 now.

In each case, make sure the chip is oriented correctly before soldering it in place. Check that IC1-IC3 are the OPA1662 op amps, IC5 is the 24LC32A and IC6 is the PIC12F1571 (if used). For each device, solder one pad first and check its alignment. Readjust the component positioning by reheating the solder joint if necessary before soldering the remaining pins.

Next, install the resistors (use a DMM to check their values), followed by the ferrite bead (FB1). Use a resistor lead off-cut to feed through the bead and solder to the board. Push the bead fully down so that it sits flush against the PCB before soldering its leads, so it doesn't rattle later. Diode D1 can be installed next. Take care to orient it correctly.

The MKT and ceramic capacitors can now go in, followed by the electrolytic capacitors. The electrolytics are polarised, so they must be oriented with the correct polarity; the longer lead goes into the hole marked with a '+' symbol.

Install potentiometers VR1-VR7 (and VR8 if used), noting that VR4 is 100kΩ and the remainder are 10kΩ. The 10kΩ potentiometers may be marked as '103', while the 100kΩ pot may be marked '104'.

Crystal X1 can now be fitted, along with CON5, the 6-way header EEPROM programming connection. Next, mount REG1 with its leads bent over so that the regulator body lies above VR4. Make sure it does not lean so far as to make contact with the metal parts of VR4. A 45° angle to the PCB face will prevent contact with the enclosure and VR4's body.

Also, now install the PC stake at the GND test point, and the two-way polarised header for the battery lead (CON4). Follow by fitting the two jack sockets (CON1 and CON2) and the DC socket, CON3.

Switch S1 must be mounted so that the lever is captured under the front sleeve contact of jack socket CON2. Note that we have provided slotted holes so that the switch can be inserted and slid along until the lever slips under the contact.

Check that the switch is open-circuit between the two outside pins when there is no jack plug inserted, and closed between the two outer pins when a jack plug is inserted. The lever might need to be bent a little so that the switch works reliably, switching at the centre of the travel between the open and closed position of the CON2 jack contact.

Mount foot switch S2 and rotary switch S3 (if used) now. Make sure these are seated fully and not skewed before soldering.

We recommend leaving the LEDs until later, when the PCB is mounted in the case.

The next step is to cut the battery wires to 60mm, then crimp or solder them to the plug pins. Insert these pins into the plug shell, making sure you get the red and black wires in the correct position. When you plug it in, the red wire should go to the terminal marked '+' on the PCB, adjacent to D1's anode.

It's necessary for the GND terminal on the board to be connected to the case, to prevent hum injection via the enclosure. Cut a 50mm length of green medium-duty wire, solder a solder lug to one end and the other to the GND terminal on the PCB. It's a good idea to place some heatshrink tubing over the lug terminal and the GND PC stake.

When assembled, the solder lug is secured to the case using an M3 x 6mm screw, star washer and M3 nut.

Powering up and testing

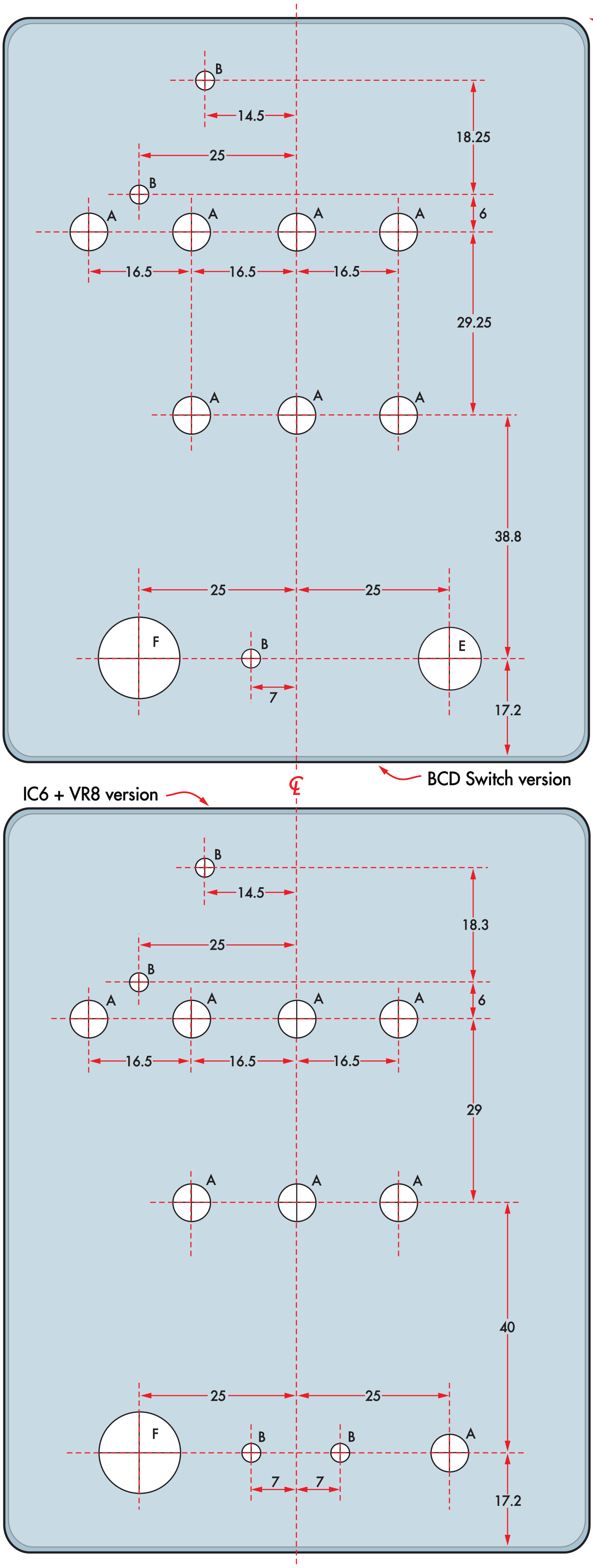
If you are planning to use a battery, connect this now. Alternatively, connect a DC supply (9-12V DC). Plug a jack lead into CON2 to switch on the power. Then, using a multimeter set to read DC volts, connect the negative probe to the GND point and measure the regulator input and output voltages.

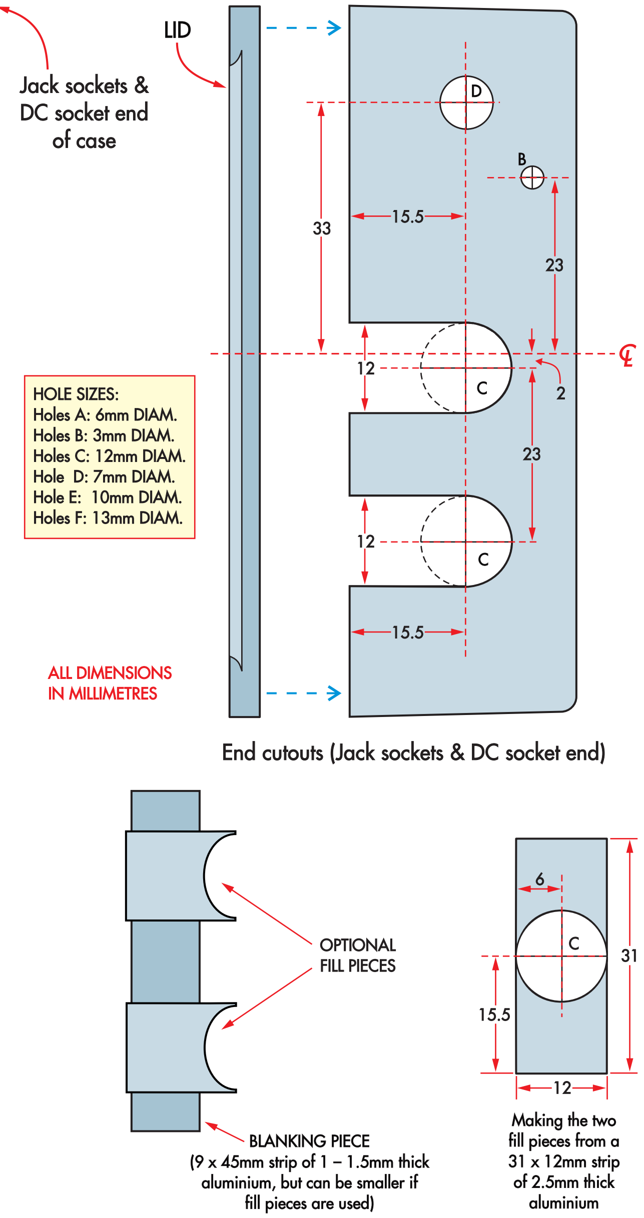
The input should be about 0.3V below the battery or DC supply voltage. The regulator output should be between 3.267V and 3.333V.

If that checks out, you can connect up a signal source and some sort of amplifier, fiddle with the knobs, and check that they appear to be working as intended.

Housing

The PCB is housed inside a 119 x 94 x 34mm diecast aluminium enclosure. We use the lid as the base, with the



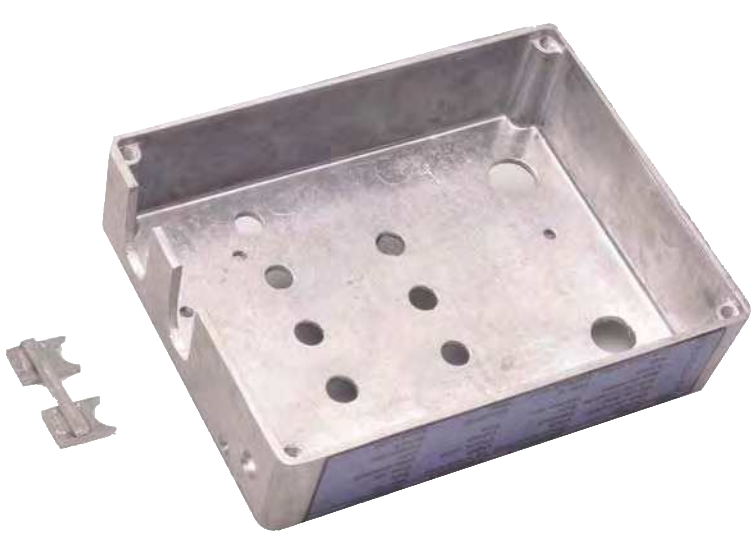


HOLE SIZES:
Holes A: 6mm DIAM.
Holes B: 3mm DIAM.
Holes C: 12mm DIAM.
Hole D: 7mm DIAM.
Hole E: 10mm DIAM.
Holes F: 13mm DIAM.

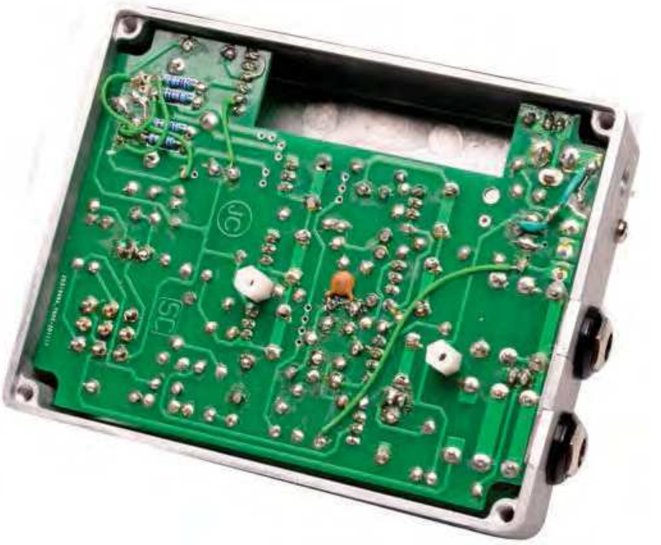
Reproduced by arrangement with
SILICON CHIP magazine 2022.
www.siliconchip.com.au

Fig.5: same-size drilling diagrams for both the mechanical switching version (top left) and the potentiometer switching version (lower left). End drilling and blanking, or infill pieces are the same for both versions. These diagrams can be downloaded from the April 2022 page of the PE website.

An upside-down view of the finished project: the box base becomes the front panel (with appropriate label) and the box lid, with four nylon screws used as feet, becomes the base. Labels fixed to each side make effect selection simple.



There are quite a few holes to be drilled in the diecast box – see the drilling template (Fig.5, opposite) for details. Note also the 'infill', or blanking, piece – this helps seal the box after the PCB is placed in it.



And speaking of placing the PCB, this photo shows how it's done Ignore the tacked-on components in our prototype: PCBs have these additions already made. Note, though, the four resistors top left are required in the switched version.

controls protruding through the main enclosure body. Use the drilling template (Fig.5) to make the required holes in the base. You can also download this as a PDF from the April 2022 page of the *PE* website.

The only differences for the two versions are that the board with a potentiometer needs an extra 3mm hole for LED4, and the shaft hole is 6mm rather than 10mm.

Cut-outs are also required in the side for the two jack sockets and DC power socket. The template shows the slots required for the jack sockets so they can be slid in place.

The resulting gaps in the side of the enclosure, after the jack sockets are inserted, can be filled in. These can be covered with a small blanking piece made from a 45mm x 9mm piece of 1mm-thick (or up to 1.5mm) aluminium. You can also glue shaped plastic or aluminium 'infill' pieces to the rectangular backing piece for the neatest possible appearance, as shown in Fig.5.

If doing this, cut a piece 31 x 12mm or a little larger, then drill a 12mm diameter hole in the centre. Once carefully filed, the piece will break apart so there will be two pieces that match the gaps in the enclosure.

For the enclosure feet, you can stick rubber feet on the 'lid'. Alternatively, you can replace the original lid-securing screws with nylon M4 screws. The nylon screw head then acts as the feet. To allow this, the holes in the enclosure for the original mounting screws will need to be drilled out to 3.5mm, and tapped using an M4 thread tap.

Panel labels

The front and side panel label artwork is also available for download from the April 2022 page of the *PE* website. The two side panels show the effects available (1-8 and 9-16). These can be affixed to the sides of the enclosure. Note that there are two front panel labels and you need to select the one which suits your build (rotary switch or pot).

A rugged front panel can be made using overhead projector film, with the label printed as a mirror-image so the ink will be between the enclosure and film when affixed. Use projector film suitable for your printer (either inkjet or laser) and affix it using clear neutral-cure silicone. Roof and gutter silicone is suitable.

Squeegee out the lumps and air bubbles before the silicone cures. Once cured, cut out the holes through the film with a hobby or craft knife.

Final assembly

Attach the 9mm-long M3 tapped spacers to the underside of the PCB. These are located just behind CON1 and CON2, and between VR5 and VR6. Secure them using an M3 screw from the top of the PCB. The spacer keeps the PCB in place by resting on the lid when the case is assembled. For the version using VR8, there is another 9mm M3 tapped spacer required near VR8.

The ground lug mounting position is adjacent to the DC socket. This is secured using an M3 screw, star washer and nut before the PCB is inserted into the case. Have the solder lug oriented so that the wire is closest to the enclosure base, so it does not foul the components on the PCB.

Before mounting the PCB in the enclosure, insert the LEDs into the PCB (longer leads to anode pads, marked 'A'). Place the nylon washers for the footswitch onto its shaft before inserting the PCB into its position in the enclosure. Then feed the LEDs into the bezels to capture them. Solder the LED leads from the rear of the PCB and trim them.

The battery compartment is the rectangular cut-out on the PCB. The battery can be prevented from moving with some foam packing sandwiched between the end of the battery and the PCB's edge. If you are not using the battery option, remove or fully insulate the battery clip at CON3 to prevent the contacts shorting onto a part of the circuit.

Knobs

Since the potentiometer shafts do not protrude much more than 9mm above the panel, standard knobs with a skirt to cover a potentiometer securing nut will not have sufficient internal fluting length to keep the knobs secured. So use knobs that don't have the skirt, as listed in the parts list.

For the PCB version that uses the rotary switch, you will need to cut the switch shaft, leaving sufficient length for the knob to attach securely close to the panel. Also, a flat will need to be filed on the side of the shaft to form a D-shape suitable for the knob. This will need to be carefully filed so it is a tight fit. The knob pointer will also need to be prised off and oriented correctly.

Knob pointer orientation is best found during the testing procedure. While 15 of the 16 positions will give an effect, position six is the test position, and the output signal closely matches the input signal. With the knob rotated to this position, adjust the pointer to line up with 6.

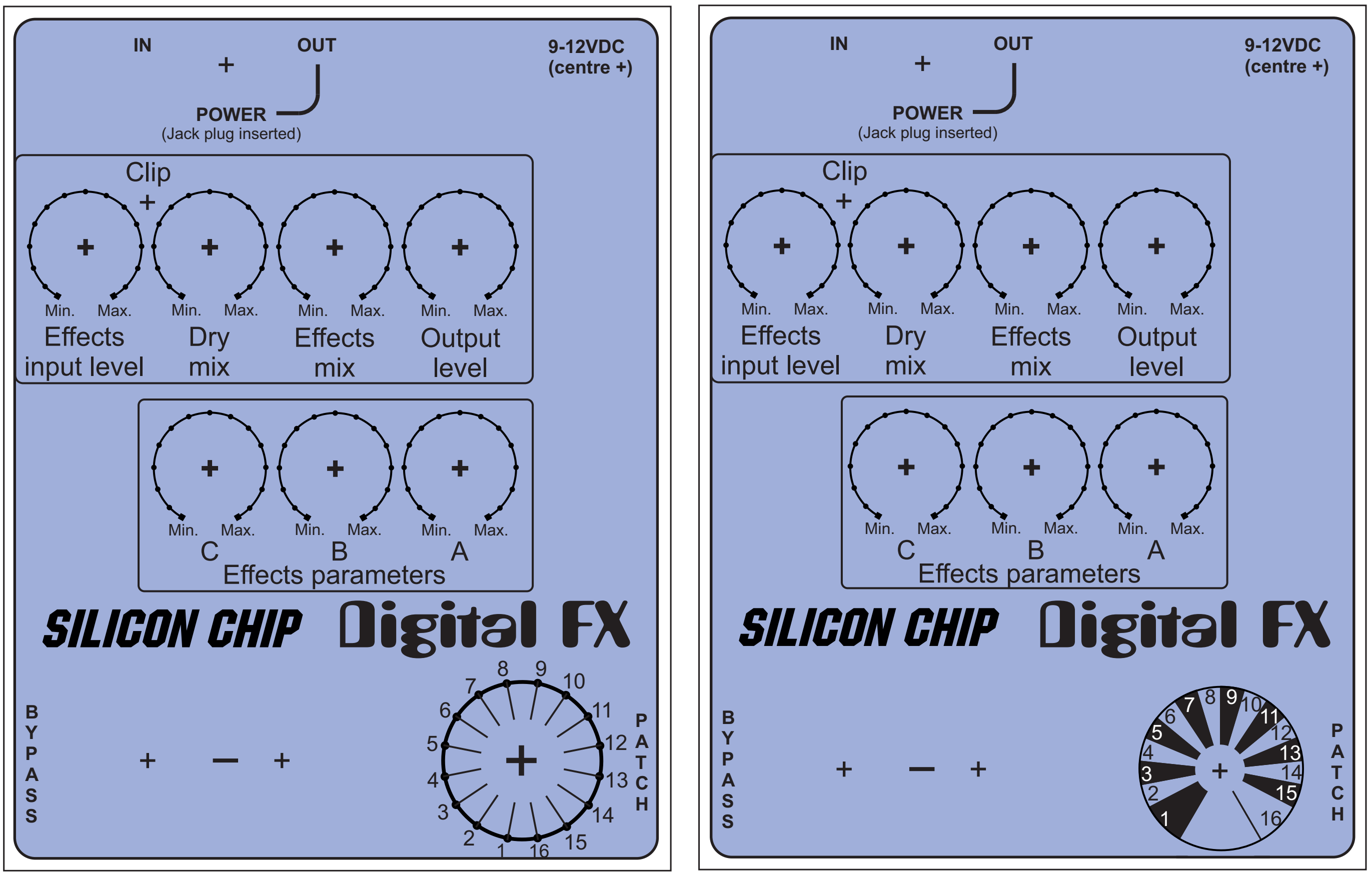


Fig.6 (above): front panels for the two versions of the project – on the right is the potentiometer-selected version while the left panel is for the switch selected. This artwork can be downloaded from the April 2022 page of the PE website.

Patch	Effect	Adjustment C	Adjustment B	Adjustment A
1	Chorus-reverb	Chorus mix	Chorus rate	Reverb mix
2	Flange-reverb	Flange mix	Flange rate	Reverb mix
3	Tremolo-reverb	Tremolo mix	Tremolo rate	Reverb mix
4	Pitch shift	-	-	+/- ~4 semitones
5	Pitch echo	Echo mix	Echo delay	Pitch shift
6	Test	-	-	-
7	Reverb 1	Low filter	High filter	Reverb time
8	Reverb 2	Low filter	High filter	Reverb time

Patch	Effect	Adjustment C	Adjustment B	Adjustment A
9	Octaver	Down octave level	Up octave level	Dry mix
10	Pitch shift glider	Glide	Depth	Rate
11	Oil can delay	Feedback	Chorus width	Time rate
12	Soft clip overdrive	Tone	Volume	Gain threshold
13	Bass distortion	Dry/wet mix	Tone	Gain
14	Aliaser	-	-	Sample rate
15	Wah	Filter Q	Sensitivity	Reverb
16	Faux phase shifter	Feedback level	Time	Speed width

Fig.7 (right): the side labels are identical for both versions and show at a glance what the various combinations achieve. Label 1-8 should be fixed to one side and label 9-16 to the other side.

Another way to this is to measure the voltage at the A, B, C and D points at pins 16, 17, 18 and 13 of IC4 respectively, when powered up. Position 1 is when all of these are at 0V.

Finally, secure the lid in place using either the original screws or nylon M4 screws, as mentioned previously. Stick rubber feet to the base if you are not using the nylon screws as 'feet'.

Removing the knobs

After installation, the knobs are likely to be difficult to remove. You will need to lever them off; make sure the lever (such as a flat-bladed screwdriver) is against a packing piece placed on the front panel to prevent damage to the panel.

Usage

Note that some patches available in the default selections use the A, B and C parameter adjustments, while other patches only use adjustment A. Also, some effects give you control over the effect/dry mix while others do not. See the side panel labels (above) for details.

When the effects parameters include a mix control, the main dry mix control should be set fully anticlockwise, the effects mix control set fully clockwise, and the mixing done with the parameter mix control(s).

Where an effect has no mixing control, the dry mix level adjustment provided can be used instead.

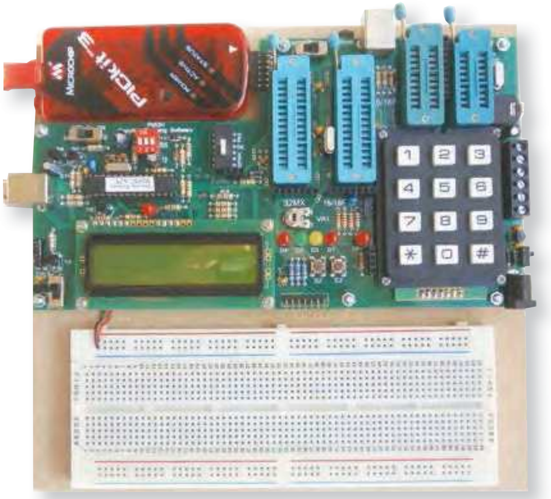
When connecting to an amplifier, it should have a switch that allows the jack's shield connection to be either earthed or floating. A guitar with piezo pickups should have less hum when the switch is selected to connect to earth.

Next month

We'll have a follow-up article next month that describes how to create and load your own effects into the EEPROM chip, changing the nature of effect selections 8-15. This can be done using freely available software and a Microchip PICkit 2 or PICkit 3 programmer.

P955H PIC Training Circuit

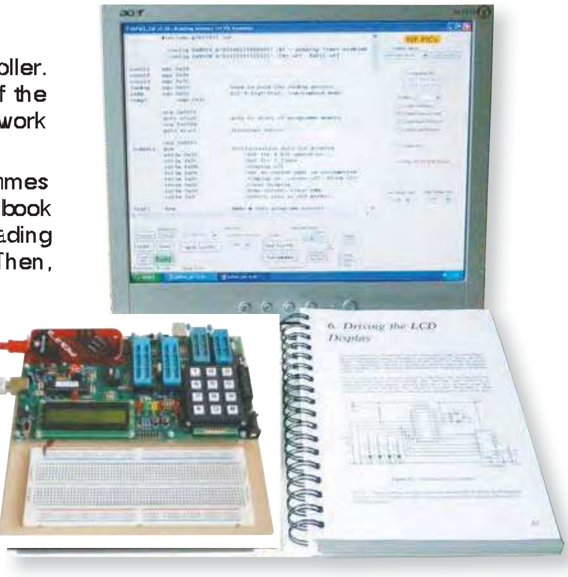
by Peter Brunning



When you are first learning about PICs, whether you are a complete beginner or an experienced programmer, you need an uncomplicated system which allows you to learn without getting bogged down in system procedure. That is why we created the P955H PIC training circuit and our own PIC assembler. In the first book we learn about PIC programming using the Brunning Software PIC assembler BSPWA, but in chapter 3M there is an introduction to the Microchip assembler MPASM X. All our assembler text will run in both systems, so from there on, if you wish you can use MPASM X. Likewise, we start by using the on-board PIC programmer to write the code into the PIC, but if you prefer, plug in a PICkit 3 or 4 and use that. The P955H training circuit has the flexibility to be what you need as your learning process advances.

The P955H training circuit has been designed to work with both 32-bit and 8-bit PICs. The idea is to start learning about PICs using assembler with 8-bit PICs. Then learn C with 8-bit PICs, study serial communications using 8-bit PICs, and finally study C programming using 32-bit PICs. It is a simple approach to a subject that has no limit to its ultimate complexity.

The Brunning Software P955H PIC Training Course



We start by learning to use a relatively simple 8-bit PIC microcontroller. We make our connections directly to the input and output pins of the chip and have full control over the internal facilities of the chip. We work at the grassroots level.

The first book teaches absolute beginners to write PIC programmes using assembler, which is the natural language of the PIC. The first book starts by assuming you know nothing about PICs, but instead of wading into the theory we jump straight in with four easy experiments. Then, having gained some experience, we study the basic principles of PIC programming, learn about the 8-bit timer, how to drive the alphanumeric liquid crystal display, create a real-time clock, and experiment with the watchdog timer, sleep mode, beeps and music. Then there are two projects to work through. In the space of 24 experiments, two project and 56 exercises we work through from absolute beginner to experienced engineer level using the latest 8-bit PICs (16F and 18F).

The second book introduces the C programming language for 8-bit PICs in very simple terms. The third book, *Experimenting with Serial Communications*, teaches Visual C# programming for the PC so that we can create PC programmes to control PIC circuits.

In the fourth book, we learn to programme 32-bit MX PICs using fundamental C instructions. Flash the LEDs, study the 16-bit and 32-bit timers, write text to the LCD, and enter numbers using the keypad. This is all quite straightforward as most of the code is the same as already used with the 8-bit PICs. Then life gets more complex as we delve into serial communications with the final task being to create an audio oscilloscope with advanced triggering and adjustable scan rate.

The complete P955H training course is £259, which includes the P955H training circuit, four books (240 × 170mm, 1200 pages total), six PIC microcontrollers, PIC assembler and programme text on CD, two USB-to-PC leads, a pack of components, and carriage to a UK address. (To programme 32-bit PICs you will need to plug in a PICkit 3 or 4. You can buy PICkit 4 from Microchip for around £50).

Prices start from £175 for the P955H training circuit with Books 1 and 2 (240 × 170mm, 624 pages total), two PIC microcontrollers, PIC assembler and programme text on CD, USB-to-PC lead, and carriage to UK address. (PICkit 3 or 4 not needed for this option.) You can buy Books 3 and 4, USB PIC, 32-bit PIC and the components kit as required later. See the Brunning website for details: **www.brunningsoftware.co.uk**

Mail order address:

Brunning Software

138 The Street, Little Clacton, Clacton-on-sea,
Essex, CO16 9LS. Tel 01255 862308

Care for your rechargeable batteries

High-current Battery Balancer

Part 1 – by Duraid Madina

Proper balancing is critical for a long battery life, especially if they are lithium-based rechargeable types. But many balancers are inefficient, as they dump excess charge for a given cell, restricting how fast you can charge the batteries and wasting power. Not this one – it redirects that extra charge into other cells, so you can charge fast with little heat or waste!

Most rechargeable batteries consist of an array of nominally identical cells, connected in series, parallel or series/parallel to meet particular voltage, current and capacity requirements.

Batteries with many series-connected cells often only expose the connections at the extreme ends. For example, a typical lead-acid car battery has six cells (2V × 6 = 12V) but only two terminals.

To charge such a battery, we apply a higher voltage than the total of all the cells across those two terminals, and current flows through all six cells, increasing their state of charge.

But there is no guarantee that each cell starts with an identical voltage, and despite their identical construction, cell capacity can vary, especially as the battery ages.

This is not a big problem with car batteries because lead-acid cells tolerate slight overcharging well. By overcharging the battery a little, cells with a lower charge get a chance to 'catch up' to the others, while the most highly charged cells dissipate the charging current as heat.

Despite this, large lead-acid battery banks (as might be used in a renewable energy installation) will last longer if they are kept balanced. In this case, you might have several batteries in series, so not only do you need to be concerned about inter-cell balancing within a given battery, you also need to consider balancing the charge between batteries.

The fact that you might be using batteries with different ages and possibly even from different manufacturers makes this even more critical.

Then there is the case of lithium-ion and similar rechargeable cells. There is a great variety of lithium chemistries around, but many of them do not tolerate overcharging. They also can be easily damaged by over-discharging. So keeping lithium rechargeable batteries balanced is even more crucial.

This *Battery Balancer* can handle cell voltages as low as 3V and as high as 15V, so it is suitable for a wide range of balancing tasks, including balancing the cells within a lithium-ion battery, or balancing individual lithium-ion or lead-acid batteries.

Each *Balancer* can handle up to four cells (or groups of cells) or batteries, and you can combine multiple *Balancers* for larger installations.

Avoiding cell damage

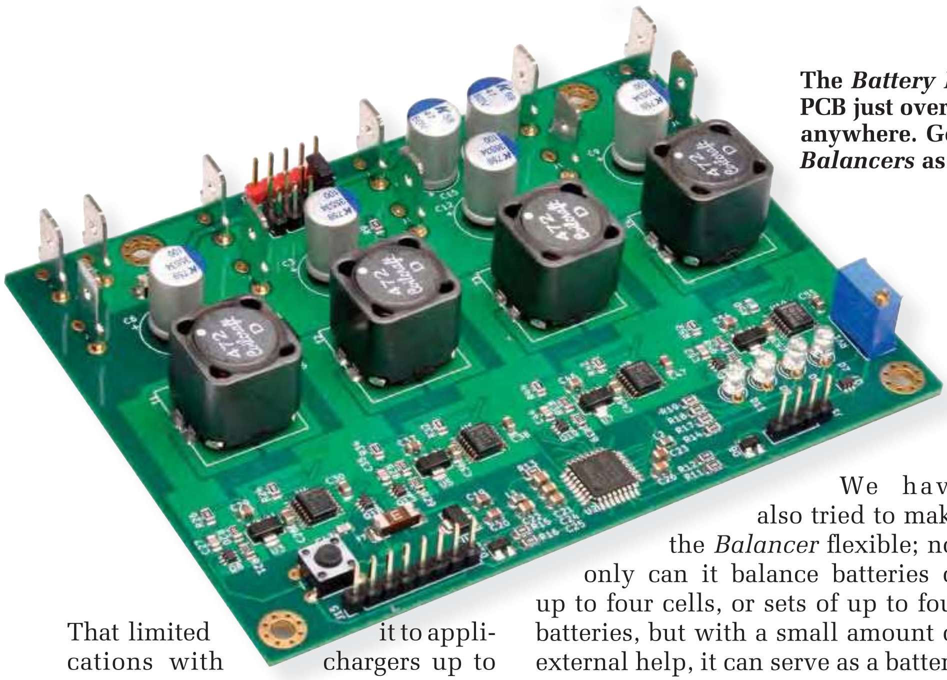
One conservative option would be to immediately stop charging as soon as any cell reached its maximum permissible voltage, but that would leave the remainder of the cells not quite fully charged.

Left unchecked, what might start as a minimal imbalance between the cells, could over repeated charge/discharge cycles develop into a much larger imbalance, with the result that as a whole, the battery has significantly less usable capacity. Worse, when the battery is fully discharged, those cells which were not fully charged could become over-discharged and damaged.

So we need a way to ensure that as a battery is charged and/or discharged, the cells are kept in balance. Each is then charged to approximately the same voltage, so that the battery capacity remains good and the cells degrade equally. This way, a battery need not be discarded just because one cell has degraded more rapidly than the others (a common problem).

The simplest way to do this is to shunt current around any cell that has a higher voltage than the others during charging. We have used that approach in the past, for example, in our April 2017 *Battery-Pack Cell Balancer.*

That design could handle packs with up to six cells, but only provided about 200mA of balance current.



The *Battery Balancer* is constructed on a single 4-layer PCB just over 100mm wide, so it's small enough to slot in anywhere. Got more than four batteries? Build as many *Balancers* as you need!

That limited it to applications with chargers up to 10A, and it got quite warm during operation, as all that power was being turned into heat.

Our new *Balancer*, being much more efficient, produces much less heat for a given balance current and thus can handle much higher battery charge currents – to 50A or more, assuming the cells are matched to within 5% (a fairly conservative figure for a healthy battery).

Operational overview

This *Battery Balancer* helps to ensure that cells in a battery are kept in balance by periodically checking the cell voltages and moving charge from cells at a higher voltage to cells at a lower voltage.

To do this, it has three main sections, as shown in Fig.1. These are:

1) A voltage sensing front-end which draws very little current from the cells.

2) A control section consisting of little more than a Microchip SAM-L10 32-bit microcontroller, which also draws hardly any current when idle (according to Atmel, it's the 'industry's lowest power in its class').

3) A power section for moving charge between cells.

The *Balancer* has been designed to achieve a high level of practical efficiency in three ways. First, the amount of power consumed when not actively balancing cells is kept low, by allowing virtually everything to be switched off, and ensuring that most of what remains draws very little power.

Second, the amount of power required to see if balancing required is kept low, through the use of simple but energy-efficient voltage dividers.

Third, instead of using inefficient schemes for balancing such as simply dumping charge from cells that have too much charge into resistive loads, the *Balancer* recycles charge by taking it from cells that have too much, and adding it to those that have too little.

We have also tried to make the *Balancer* flexible; not only can it balance batteries of up to four cells, or sets of up to four batteries, but with a small amount of external help, it can serve as a battery charger and even a battery discharger!

Transferring charge

Perhaps the most critical part of the *Battery Balancer* is the section which transfers charge between batteries/cells (and maybe also a charger or a load). This section is replicated four times on the board, once for each battery/cell that can be connected.

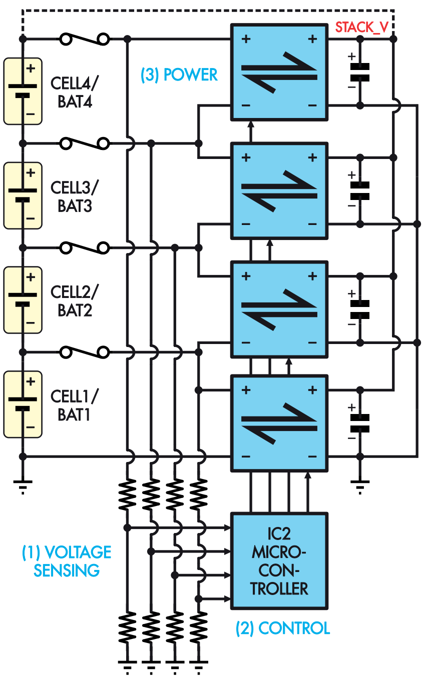


Fig.1: while highly simplified, this shows the basic configuration of the *High-current Battery Balancer*. Microcontroller IC2 measures the voltage across each battery/cell via resistive dividers. If one has a voltage that is significantly higher or lower than the others, it transfers power into or out of the imbalanced cells via the four power transfer blocks. These can efficiently transfer energy to or from one battery/cell to the entire 'stack', and by extension, between multiple batteries/cells via the stack.

A simplified version of this circuit is shown in Fig.2.

This section can transfer energy to or from the battery/cell shown at the left and the complete battery/'stack'. Energy can be transferred from one battery/cell to another via the 'stack'.

Let's suppose we notice that one cell has a voltage that is lower than the other three. We can use that cell's power section to transfer charge from the whole battery to that cell, to bring it into balance. This happens cyclically.

First, the 'stack-side' transistor (QX) is switched on and current begins to flow from the battery, through the stack-side transformer winding, energising the transformer's core. Because the cell-side power transistor (QY) is off, the voltage across the cell-side transformer winding quickly rises.

A moment later, the stack-side power transistor is switched off, and the cell-side power transistor (QY) is switched on. This transfers the energy from the transformer's core into the cell. When this is estimated to have completed, the cell-side power switch is turned off and the cycle repeats, with a duty cycle proportional to the desired rate of charge transfer.

The inductance of the transformer can be chosen relatively freely. Transformers with higher inductance allow operation at lower frequencies, but have higher resistive losses. Transformers with lower inductance require operation at higher frequencies, but have lower resistive losses.

Note, however, that transformers with particularly low winding inductances tend to have slightly reduced coupling between the windings, though only a few such transformers have coupling so poor as to be a significant factor for this *Battery Balancer*.

The voltages at the drains of QX and QY can exhibit significant inductive ringing. If it is too severe, it might exceed the transistor ratings.

We have attempted to keep the inductance of these paths low by placing these devices very close to their respective transformers. But for higher-voltage applications, it is still prudent to place series RC snubbers (ie, Csnub and Rsnub) across the transformer windings.

For lower-voltage applications (eg, balancing lithium-ion cells), these snubbers can be safely omitted, and that might even result in a small efficiency gain.

The micro controls MOSFETs QY and QA via an ISO7041 isolator because the negative end of the battery/cell is not connected to ground (unless it is the bottom-most in the stack). The driving scheme is a bit more complicated than shown here, as will soon become apparent.

The ISO7041 is powered by its own 'floating' 3.3V regulator from the battery/cell, to allow for the battery/cell voltage to vary over a wide range.

Note how the negative terminals of the bypass capacitors both for the individual battery/cell and for the stack are connected via N-channel MOSFETs, rather than directly to the negative terminal of the battery/cell and GND respectively. This is to provide a 'soft-start' function which greatly reduces the sparks generated when connecting up batteries or cells.

Features and specifications

- Balances two, three or four series-connected cells or batteries
- Suits Li-ion, LiPo, LiFePO4, lead-acid, AGM and other chemistries
- Each cell or battery can range from 2.5V (fully discharged) up to 15V maximum
- Balancing current: up to 2.5A
- Charging current: up to 50A
- Efficiency: typically around 80%
- Quiescent current: around 100µA per battery/cell
- 5mm spade lug connections for high-current batteries
- 2.54mm-pitch pin header for connecting smaller batteries
- Switching frequency: typically 100kHz
- Multiple Balancers can be combined for balancing more cells or batteries
- It can also act as an efficient battery charger or discharger
- Four onboard status LEDs plus one adjustment potentiometer
- Serial status/debugging interface
- Compact size (108 x 80mm PCB)

Full circuit details

The full *Battery Balancer* circuit is shown in Fig.3, although two of the four charge-balancing circuits have been partly omitted to save space. All four are configured identically. Now you can see the full detail of this part of the circuit, which reveals a few extra subtleties.

First, the isolator outputs cannot drive the MOSFET gates and microcontroller directly as they are too weak to achieve the required switching speeds. We spent a lot of time investigating the use of integrated gate-driver ICs in that role, but most of them have a significant quiescent current draw and stop functioning at low supply voltages.

While this could be resolved for the lower cell power sections by deriving their supply rails from 'one cell up', this would leave the topmost power section needing an alternative source of power, eg, from a boost converter.

Instead of using integrated gate drivers, we decided instead to use simple NMOS/PMOS transistor pairs configured as inverters. Happily, there are many dual SMD MOSFETs available which include one N-channel device and one P-channel device, so each inverter is contained within a single package. In the case of the uppermost section of the circuit, these are Q11 and Q12.

In each case, the MOSFET driving the stack side of the transformer (eg, Q9) is connected source-to-ground, and is a logic-level FET. It is driven by the 0-3.3V output of the inverter pair, which are themselves driven from a microcontroller digital output pin (in this case, pin 22, labelled SSPWM3).

A 10kΩ pull-up resistor is provided at the input of each of these MOSFET-driving inverters so they have a low output when the micro is not in control of that pin (eg, it is in reset or being programmed).

The other transformer-connected MOSFET (for example, Q10) has its source connected to the junction of this battery/cell and the one below. So as we described above, it is driven by an isolator that runs off a 3.3V floating supply referenced to that same voltage. Therefore, the MOSFET-driving inverter is also connected across this 3.3V floating supply, to provide an appropriate swing for that MOSFET.

It too has a 10kΩ pull-up resistor to hold the MOSFET off by default. But note that the Texas Instruments ISO7041 low-power digital isolator has variants with different default pin states. The one we have chosen provides high outputs if its inputs are not driven, or the input side of the device is not powered (as opposed to the ISO7041F, which offers low outputs). This provides us with a safe 'resting' state.

1Ω resistors limit the power through the gate drive inverters, adding to the inverters' intrinsic ~0.2Ω output resistance. This keeps the peak gate drive currents below 3A.

It is not critical that the low-dropout (LDO) floating regulator (REG3 here) falls out of regulation if the cell voltage drops below 3.3V, as both the isolator and gate driver are capable of operating below this voltage.

Note though that if a cell voltage is ever at less than 2.5V (a dangerously low voltage for a lithium-polymer cell, and a very low voltage for a lithium-ion cell), no attempt will be made to transfer charge to or from this cell.

Instead, it is assumed that a battery with cell voltages this low is likely to have minimal charge, and so even if imbalanced, merely charging the entire battery will quickly bring the cell

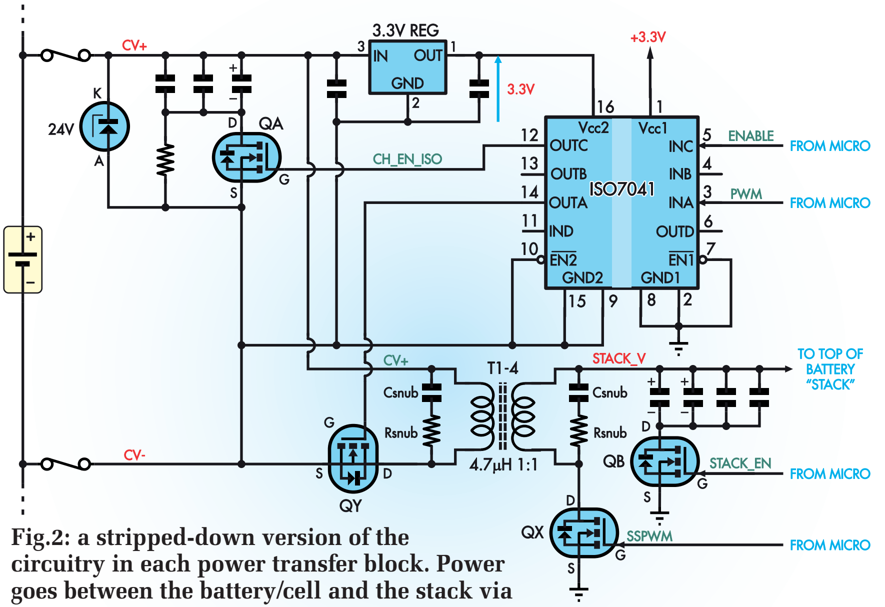
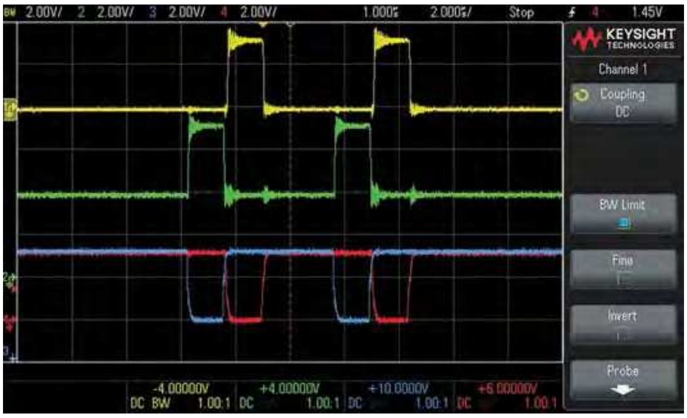
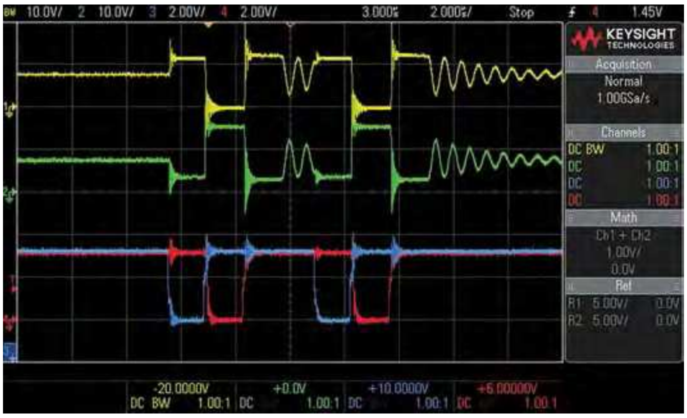


Fig.2: a stripped-down version of the circuitry in each power transfer block. Power goes between the battery/cell and the stack via MOSFETs QX and QY and the transformer. QX is ground-referenced, so it is controlled from a microcontroller output pin, while QY is referenced to the negative cell/battery terminal. Therefore, the signal from the microcontroller to control QY goes through an ISO7041 isolator, which is powered from a 3.3V rail derived from the cell/battery voltage.



Scope1: this shows how the power switching MOSFETs are driven. For clarity, two isolated pulses are shown. The red and blue traces show stack-side and cell-side PWM signals for balancer channel 2 (gate-driving inverter inputs) as driven by the microcontroller and digital isolator, respectively. The yellow and green show the stack-side and cell-side power MOSFET gate voltages (gate-driving inverter outputs). The majority of the ringing on these traces is due to measurement error.



Scope2: here, the red and blue traces are as in Scope 1, but the yellow and green traces show the drain voltages of the main switching MOSFETs (ie. the bottom ends of the power transformer) – the stack-side node is in yellow, while the cell side node is in green. Here, less of the ringing on the switching nodes is due to measurement error, particularly in the phase where both power MOSFETs are off, allowing their drains to float.

voltages above 2.5V. Balancing can then resume long before any of the cells approach full charge.

For more details on how charge transferral works, refer to scope grabs Scope1 and Scope2 and their captions.

Voltage sensing

To know which batteries or cells should be charged or discharged, the *Battery Balancer* needs to be able to take accurate voltage measurements across each battery/cell.

Sensing low voltages accurately is becoming easier; high-performance analog-to-digital converters (ADCs) are readily available, and modern microcontrollers often include ADCs that would have been considered high-performance not that long ago. In our case, the SAM-L10 micro has a 12-bit ADC capable of taking one million samples per second.

As we need to sense voltages up to around 60V (say, four 12V lead-acid batteries in series under charge), a kind of front-end is required to bring these voltages down into typical ADC ranges.

One option would be to use operational amplifiers (op amps) that can tolerate these higher voltages, to divide (and possibly shift) the voltages as required. Suitable parts are not hard to find, but they are not cheap.

Moreover, because the voltages the *Battery Balancer* needs to sense do not vary quickly, very little in the way of high-frequency performance is required, so offset-correcting chopper-style op amps are applicable.

However, the performance of the required op amp circuits would be dominated by the accuracy of the connected resistors. The power consumption of these op amps, while impressively low in many devices, is high enough that we couldn't leave them powered all the time.

So instead, we use a simple switched-capacitor, switched-ground resistive voltage divider, as shown in Fig.3. To avoid the constant power consumption of an always-on voltage divider, we add low-side NMOS FETs (Q8a, Q13a, Q19a and Q24a). Even very small-signal FETs introduce only a couple of ohms of error while on.

When off, however, the voltage can drift above the tolerance of the microcontroller input pins. So a second set of NMOS pass transistors (Q8b, Q13b, Q19b and Q24b) ensures the microcontroller never sees such voltages. Once again, we can take advantage of dual MOSFET packages so that each pair of transistors is just one part to be soldered to the board.

To save microcontroller pins, all of the voltage dividers share a common pair of control lines. To take a set of voltage readings, first, the low-side NMOS switches are turned on, enabling the divider. Next, the pass gate NMOS switches are turned on, allowing the filter capacitors to start settling towards their respective values.

Finally, the microcontroller's onboard ADC takes its samples, allowing the software to know the voltage across each battery or cell. With the 100kΩ/2.2kΩ dividers used, and the 12-bit ADC having a 1.65V reference, the nominal sensed voltage range is 0-76.65V, and the resolution is 18.7mV. That's precise enough to detect small differences between 12V battery voltages.

Refer to Scope3 for more information on how this process works.

For lower voltage batteries such as li-ion, LiPo or LiFePO4 packs with cells typically ranging from 2.7-4.2V, the resistive dividers are changed to 100kΩ/6.8kΩ which gives a range of 0-25.9V and a resolution of 6.3mV, which means we can balance out inter-cell voltage differences starting at about 10mV.

A virtually identical arrangement is used to sense the voltage across the whole stack using MOSFETs Q18a and Q18b (which will probably be the same as one of the cells, but not necessarily the same one, hence the separate divider) and also the rotation of potentiometer VR1 via MOSFETs Q7a and Q7b. This is used to set various parameters, which will be described later.

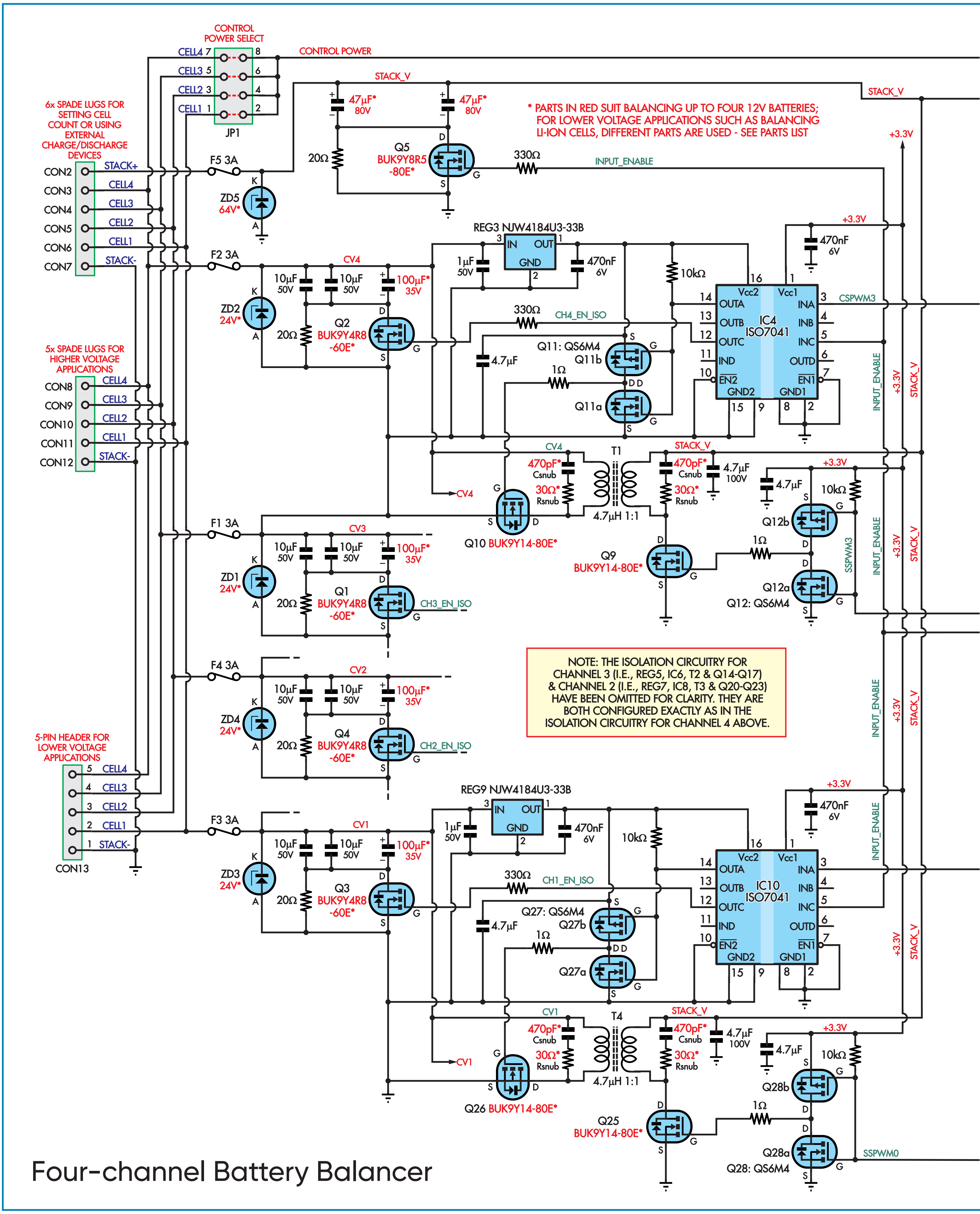
While an independent stack voltage monitor might seem redundant, it comes in handy when using the *Battery Balancer* in other applications. For example, it can be used to allow charging batteries from other power sources such as solar panels, or also as a battery charger.

It can even be used in conjunction with another *Battery Balancer*, to transfer energy between two different batteries, in either direction, while keeping both in balance.

Note that to avoid error, we don't take voltage readings while the power section is active.

Soft starting/spark mitigation

We found that the first prototype produced some nasty sparks when connecting batteries (as is not uncommon). This was mainly due to the inrush current to charge the capacitor banks. These sparks could possibly damage the connectors, or even weld them! We therefore decided that, since it was not difficult to mitigate this, we would do so.



* PARTS IN RED SUIT BALANCING UP TO FOUR 12V BATTERIES; FOR LOWER VOLTAGE APPLICATIONS SUCH AS BALANCING LI-ION CELLS, DIFFERENT PARTS ARE USED - SEE PARTS LIST

NOTE: THE ISOLATION CIRCUITRY FOR CHANNEL 3 (I.E., REG5, IC6, T2 & Q14-Q17) & CHANNEL 2 (I.E., REG7, IC8, T3 & Q20-Q23) HAVE BEEN OMITTED FOR CLARITY. THEY ARE BOTH CONFIGURED EXACTLY AS IN THE ISOLATION CIRCUITRY FOR CHANNEL 4 ABOVE.

Four-channel Battery Balancer

When power is first applied, the MOSFETs in series with the negative terminals of each set of bypass capacitors are off. Those capacitors therefore slowly charge via the parallel 20Ω resistors.

After the initial battery connection is made but before any balancing takes place, the microcontroller switches these MOSFETs on, presenting the full decoupling capacitance only after the connection is made.

The MOSFETs effectively increase the ESR of the capacitor banks a little. However, with on-resistances that are only a fraction of an ohm, the capacitors are still capable of doing their job of stabilising the cell and battery

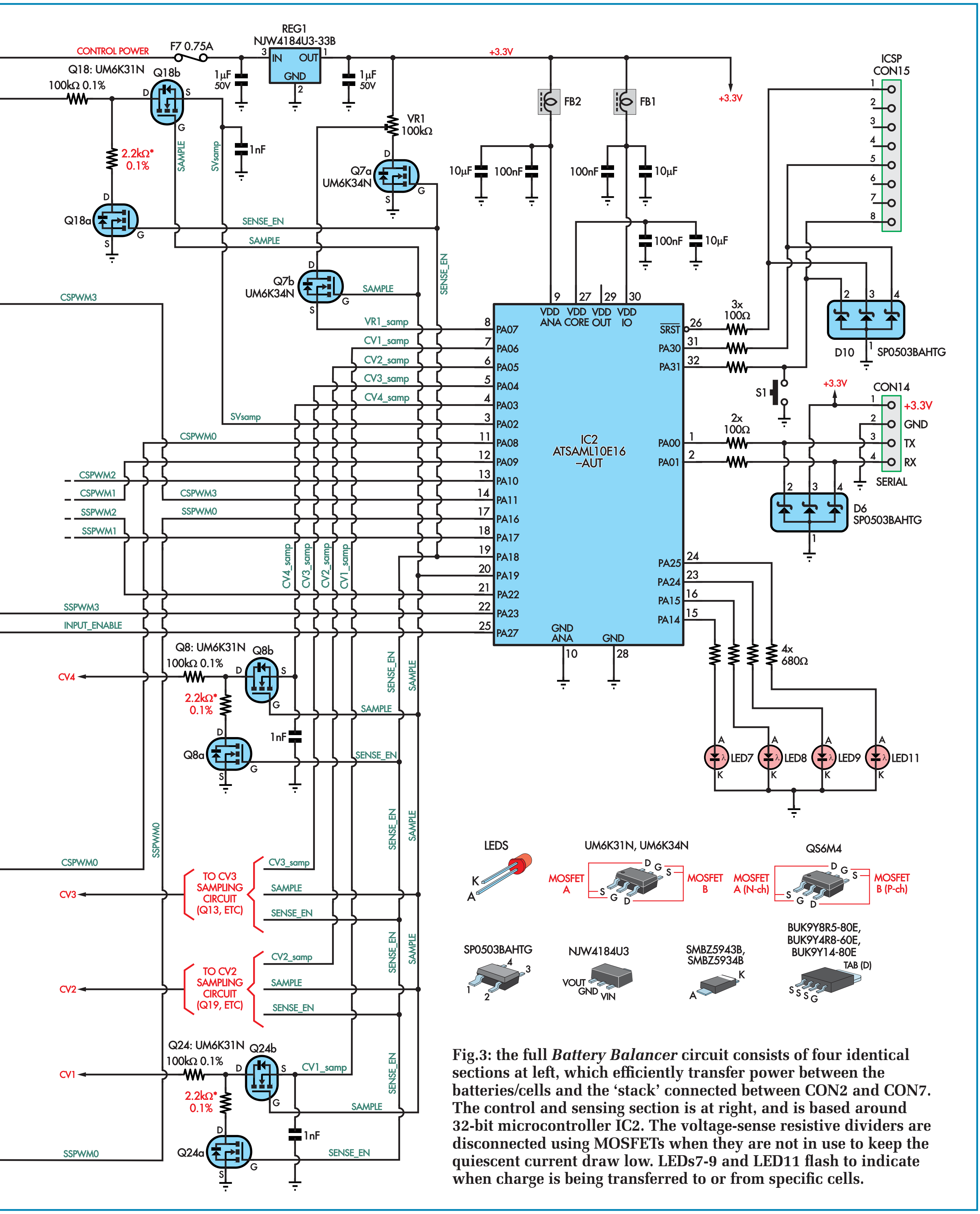


Fig.3: the full *Battery Balancer* circuit consists of four identical sections at left, which efficiently transfer power between the batteries/cells and the 'stack' connected between CON2 and CON7. The control and sensing section is at right, and is based around 32-bit microcontroller IC2. The voltage-sense resistive dividers are disconnected using MOSFETs when they are not in use to keep the quiescent current draw low. LEDs7-9 and LED11 flash to indicate when charge is being transferred to or from specific cells.

voltages nicely. The MOSFET turn-on time is quite slow because there are no inverters to drive them – however since they only switch on after the capacitors have charged, this doesn't matter.

While we still recommend taking care to positively connect batteries to the *Battery Balancer* and being prepared for some amount of sparking to take place, this approach does greatly reduce the sparking that typically occurs.

Circuit protection

No you've doubt noticed that all cell and battery connections are via fuses; a good idea given how much current a large battery (or in some cases, even a small one) can deliver if there is a fault.

Each input also has a zener diode across it (after the fuse) which provides two functions. One, if a cell or battery is connected backwards, the zener will immediately conduct and blow the fuse. Two, if the cell or battery voltage is too high for some reason (eg, you've connected to the wrong battery terminal), the zener will go into avalanche breakdown, and in most cases, the fuse will again blow.

By the way, in the parts list we've specified unidirectional transient voltage suppressors (TVSs) instead of zener diodes for these parts. They are effectively zener diodes, just with very high pulse current handling capability. Also note that the actual clamping voltage will be somewhat higher than the specified voltage, depending on the current being delivered from the source.

We have taken that into account when selecting the parts, so that the protected parts of the circuit will not be exposed to damaging voltages at any reasonable current level.

As the micro monitors all the various voltages, it will shut down if any of them are out of range. For example, if a cell voltage is too low for the circuit to function.

Control section

The microcontroller section is quite straightforward due to the high level of integration on the SAM-L10 micro (IC2).

Its internal oscillator is more than adequate as an instruction clock source in this application. Current-limiting resistors on digital outputs 15, 16, 23 and 24 are provided for it to drive four status LEDs directly (more on these later). ESD clamps are connected across the programming and UART interfaces to protect them from static discharge as these pins could be externally accessible.

The microcontroller derives its power from linear regulator REG1, another NJW4184U3-33B. This was chosen to minimise quiescent current and operate over a relatively wide input voltage range (up to 35V). Its output passes through ferrite beads before reaching the microcontroller supply pins. It also provides power to the 'near side' of the various low-power digital isolators and the stack-side gate drivers.

As these consume only a few milliamps while active, the power dissipated in the linear regulator is only a few tens of milliwatts in the worst case, when it is powered by a fully-charged 12V battery. While the gate drivers consume small amounts of current on average, they do so in an extremely bursty fashion, so they each have a local bypass capacitor.

Parts list – High Current Battery Balancer

(suitable for 12V battery balancing – see below for other options)

1 four-layer plated through PCB coded 14102211, 108 x 80mm, available from the PE PCB Service
4 4.7µH 1:1 transformers (T1-T4) [eg, Coilcraft MSD1278**]
5 3A fast-acting SMD fuses, M6125/2410-size (F1-F5) [eg, Bourns SF-2410FP300W-2]
1 0.75A fast-acting SMD fuse, M6125/2410-size (F7) [eg, Bourns SF-2410FP075W-2]
2 SMD ferrite beads, 470Ω @ 100MHz, M2012/0805-size (FB1,FB2) [eg, Taiyo Yuden BK2125HM471, Murata BLM21AG471SZ1D or Kemet Z0805C471BSMST]
1 100kΩ vertical multi-turn trimpot (VR1)
1 momentary SPST tactile pushbutton switch (S1)
11 5.08mm pitch PCB-mount vertical spade lugs (CON2-CON12) [eg, Altronics H2094/H2095]
1 5-pin straight or right-angle header (CON13; optional – for smaller battery packs)
1 4-pin header (CON14)
1 8-pin header (CON15; optional, for ICSP)
1 2x4-pin header (JP1)
1 jumper/shorting block (JP1)

Semiconductors
1 ATSAML10E16A-AUT 32-bit microcontroller programmed with 1410221A.hex, TQFP-32 (IC2)
4 ISO7041 4-channel digital isolators, QSOP-16 (IC4,IC6,IC8,IC10) *[Note:* ***not*** *ISO7041F]*
5 NJW4184U3-33B# 3.3V LDO regulators (REG1,REG3,REG5,REG7,REG9)
4 BUK9Y4R8-60E* NMOS FETs, LFPAK-56 (Q1-Q4)
1 BUK9Y8R5-80E* NMOS FET, LFPAK-56 (Q5)
1 UM6K34N dual NMOS FET, SOT-363 (Q7)
5 UM6K31N dual NMOS FETs, SOT-363 (Q8,Q13,Q18,Q19,Q24)
8 BUK9Y14-80E* NMOS FET, LFPAK-56 (Q9,Q10,Q14,Q15,Q20,Q21,Q25,Q26)
8 QS6M4 dual NMOS+PMOS FETs, SOT-457T (Q11,Q12,Q16,Q17,Q22,Q23,Q27,Q28)
4 3mm or 5mm through-hole LEDs (LED7-LED9,LED11)
4 SMD 24V* TVS diodes, SMB size (M3226/1210) size (ZD1-ZD4) [eg, SMBJ24A]
1 SMD 64V* TVS diode, SMB size (M3226/1210) size (ZD5) [eg, SMBJ64A]
2 5V ESD clamp diode arrays (D6,D10) [Littlefuse SP0503BAHTG]

Capacitors (all SMD M2012/0805 size X7R ceramic unless otherwise stated)
4 100µF* 35V radial organic polymer electrolytic (eg, Kemet A759KS107M1VAAE031)
2 47µF* 80V radial organic polymer electrolytic (eg, Kemet A759KS476M1KAAE045)
4 4.7µF 100V or 10µF 75V M3226/1210
11 10µF 50V
8 4.7µF 6V
6 1µF 50V
8 470nF 6V
3 100nF 50V
5 1nF 50V C0G
8 470pF* 250V C0G (Csnub)

** for lower-current applications, Coilcraft MSD1278-562 is a suitable alternative
# AP7370-33Y-13 is a suitable alternative
Note: Csnub and Rsnub components are not fitted for 4V/cell version

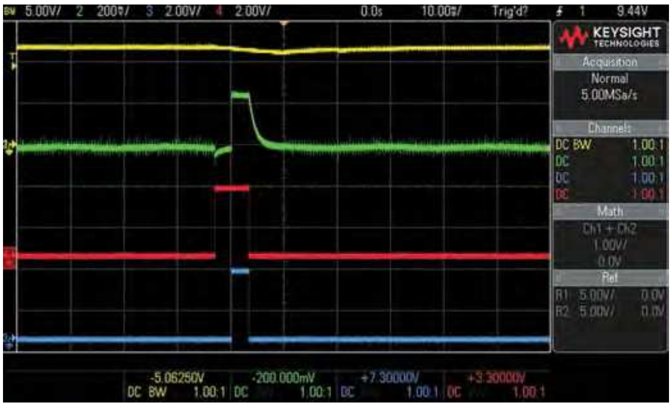
Resistors (all SMD M2012/0805 size 1% metal film unless otherwise stated)
5 100kΩ 0.1%   8 10kΩ   5 2.2kΩ* 0.1%   4 680Ω   5 330Ω   5 100Ω   5 20Ω
8 30Ω* (Rsnub)
8 1Ω M1608/0603-size

Parts for ~4V cell balancing (eg, li-ion) – substitute for asterisked (*) items above
5 BUK9Y1R3-40H NMOS FETs, LFPAK-56 (Q1-Q5)
8 BUK9Y12-40E NMOS FET, LFPAK-56 (Q9,Q10,Q14,Q15,Q20,Q21,Q25,Q26)
4 SMD 10V TVS diodes, SMB size (M3226/1210) size (ZD1-ZD4) [eg, SMBJ10A]
1 SMD 24V TVS diode, SMB size (M3226/1210) size (ZD5) [eg, SMBJ24A]
4 100µF 16V radial electrolytic polymer capacitors
2 33µF 35V radial electrolytic capacitors
5 6.8kΩ 0.1% M2012/0805 size metal film resistors

Software

The *Battery Balancer* software is fairly simple, but it took some development to get it right, and there were a few choices to be made along the way.

Perhaps the most critical task the CPU has to perform is producing the eight PWM signals required for balancing. There are many larger microcontrollers, frequently aimed at



Scope3: this shows the voltage sensing circuit in operation. The yellow trace shows the voltage to be measured (~12V) and the green trace shows the divided voltage present on the micro input pin (~240mV). The red trace is the voltage divider enable line, which has a duty cycle of less than 1%, minimising the power consumption of the voltage dividers. The blue trace is the divided voltage pass control line, which ensures that only stable divided voltages reach the micro input pin.

motor control applications, that feature large numbers of advanced PWM generators. The SAM L10 is small, inexpensive, and sips power, but has a more limited set of peripherals.

The *Battery Balancer* needs to produce short pulses of variable length at variable frequencies; if a pulse is too long, substantial currents can flow through the *Battery Balancer*, leading to a blown fuse and possibly damage to other components, particularly the power MOSFETs. Moreover, the *Battery Balancer* needs to produce two PWM signals per cell.

To achieve this, we use a software-driven approach. When a cell is to be charged or discharged, we define a ‘blip’ routine as a series of instructions that are either NOP (no-operation), or single-I/O set/clear instructions.

With a 16MHz CPU frequency, this allows us to control pulse trains with roughly 60ns precision. We then compute the desired number of ‘blips’ up to a safe maximum (currently set to 10,000), disable interrupts, and call the blip routine in a loop.

Once the blip routine has run the desired number of times, the software stops all power train activity and determines the next course of action.

Voltage sensing

When not in the middle of charging or discharging cells to bring a battery into balance, the *Battery Balancer* periodically checks the cell/battery voltages to determine which should provide charge, and which need to be given charge.

We set the ADC voltage reference to Vdd/2 (ie, around 1.65V), noting that as the power train is inactive, the power consumption and consequently noise on the Vdd LDO output will be relatively small. Therefore, this voltage should be nice and steady.

To measure a set of cell voltages, we first enable the resistor dividers by connecting their bottom ends to ground via the small-signal NFETs, and then enable the pass-transistor NFETs. We then pause for about 1ms while the capacitors on each of the sense lines settles towards their final value. Finally, we use the ADC to sample each of the settled lines before disabling the pass transistors and voltage dividers.

The rotation of potentiometer VR1 is sensed at the same time that the other voltages are measured. It can be used to configure both the peak balancing current and the cell mismatch threshold above which balancing takes place.

Serial/USB interface

The microcontroller features a UART, which is connected (via slew-limiting resistors and ESD clamping diodes) to pin header CON14. This can be easily converted to USB through the use of third-party ICs or cables such as FTDI’s ‘TTL-234X-3V3’, though note that these cables cannot be plugged directly into this header; some jumper leads will be required.

If electrical isolation is required (or at least desired), our *Mini Isolated Serial Link* project in lat month’s issue, could be connected between the *Battery Balancer* board and the USB/serial adaptor.

This board can be programmed by plugging a PICkit 4 into the ICSP header (CON15). For safety, this should only be done with no batteries or cells connected to the *Battery Balancer*.

The board features four LEDs, one for each battery/cell. These are off by default but blink slowly if a battery/cell is being charged, or rapidly if a battery/cell is being discharged. The power consumed by the *Battery Balancer’s* control logic is small compared to that consumed by the LEDs while switched on! For this reason, the LED duty cycles have been kept low.

Next month

In part two of this feature next month, we will cover building the *Battery Balancer*, testing it, configuring it and using it, as well as some safety tips.

Reproduced by arrangement with
SILICON CHIP magazine 2022.
www.siliconchip.com.au

CRICKLEWOOD ELECTRONICS
Established 1981

Your best bet since MAPLIN
Chock-a-Block with Stock

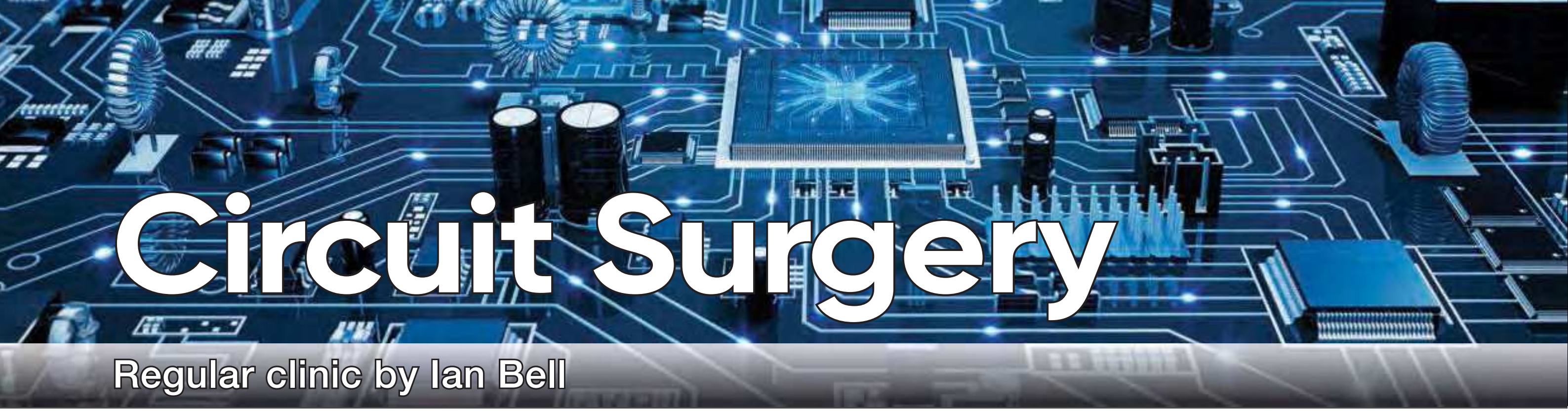
Visit: www.cricklewoodelectronics.com

Or phone our friendly knowledgeable staff on **020 8452 0161**

Components • Audio • Video • Connectors • Cables
Arduino • Test Equipment etc, etc

Visit our Shop, Call or Buy online at:
www.cricklewoodelectronics.com
020 8452 0161

Visit our shop at:
40-42 Cricklewood Broadway
London NW2 3ET



Circuit Surgery

Regular clinic by Ian Bell

The LM35 temperature sensor

In this month's article we look at the LM35 – a widely used temperature sensor integrated circuit (IC) –see this month's Flowcode article on p.49.

Temperature is one of the most frequently measured physical quantities and is commonly used as an input to microcontroller unit (MCU) based systems (and embedded systems in general). Typically, a (contact) temperature sensor produces an analogue signal (voltage or current) which varies in a well-defined way with the temperature of the sensor. This can be converted to a digital representation by an analogue-to-digital converter (ADC), which may be built into the MCU, be part of a temperature-sensing IC, which communicates with the MCU digitally; or it may be a separate converter chip. The MCU reads the ADC (directly or indirectly) to obtain a binary number related to the temperature of the sensor. Usually, this number must be processed in some way to obtain the temperature value on a standard scale, such as degrees Celsius (°C).

The complexity or processing required depends on the type of sensor used and the accuracy of measurement needed – one advantage of the LM35 is the straightforward relationship between its output and temperature being measured. Although processing is more complex if code is used to compensate for accuracy variation with temperature. Of course, analogue-output sensors can also be used in circuits which do not require an MCU or any code to be written – for example, analogue temperature switches and controllers.

In an MCU system using an analogue sensor IC, such as the LM35, the ADC, which may be on the MCU chip or external, must be set up correctly. This includes setting the reference voltages which determine the ADC's conversion range. The output voltage range from a typical temperature sensor may be smaller than the default input range of the ADC, so if the ADC range is not adjusted then the full ADC resolution will not be used. Correct ADC setup is also important when using other analogue temperature sensors.

Sensor choices

In general, when designing a temperature measurement system, a decision has to be made on what type of sensor is to be used – there are several to choose from. A broad category is contact and non-contact sensors. The most common approach to non-contact temperature sensing measures infrared radiation and includes pyrometers and thermal imaging cameras, both of which measure the temperature of the surface of the object they are 'looking at'.

Contact temperature sensors include thermistors, thermocouples, resistance temperature detectors (RTDs) and IC temperature sensors. These devices are all fundamentally based on electrical properties of materials which vary with temperature in predictable ways that can be calibrated to provide a useful measurement. The devices vary in terms of their basic physics as well as the materials and fabrication processes used to build them. The advantage of ICs over the other devices listed is in their enhanced functionality and/or simplicity of use, but they cover a narrower range of temperatures than sensors such as thermocouples and cannot achieve the accuracy of the best RTD circuits.

Temperature sensor ICs fall into two main categories – those that output an analogue signal (usually voltage) directly related to temperature and those with digital interfaces. The latter are convenient in microcontroller systems with relatively low accuracy requirements, but will require the user to manage the digital interface (such as an SPI or I^{2}C bus).

Principles of operation

Many IC temperature sensors, like the LM35, are based on the temperature-dependent behaviour of semiconductors, specifically the PN-junction found in diodes and bipolar transistors. Circuit designers often spend a lot of effort trying to overcome the effects of temperature on transistor circuits (see the December 2021 and January 2022 *Circuitry Surgery* articles on logarithmic and exponential amplifiers) – temperature-dependent signals are not hard to obtain, but the challenge when developing a sensor (like the LM35) is to obtain a linear, and well-controlled dependence between temperature and device output over a wide temperature range.

An ordinary diode (PN junction) can be used as a temperature sensor, since its forward voltage changes by approximately –2mV/°C. Improved accuracy can be obtained by using two diodes (or transistor base-emitter junctions) – the voltage difference between two PN junctions, operated at different current densities, varies linearly with absolute temperature. This temperature sensitivity has been exploited for many temperature sensor ICs, such as the LM34, LM35, LM50, LM60, LM61, MCP9700, MCP9701, TMP35, TMP36, TMP37 and TSIC301.

Other approaches are also used. For example, the AD22100 and AD22103 use a temperature-dependent resistor. The LMx35 series (x is 1,2,3) are like zener diodes with breakdown voltages directly proportional to absolute temperature at 10mV/K. The LM135's range is from –55°C to 150°C. There is also the AD950, which is a two-terminal device. This has a linear current output of 1μA/K (again, a kelvin scale) with a supply of 4V to 30V over a measurement range of –55°C to +150°C.

How the LM35 Works

We can relate the emitter current (I_E) of a bipolar transistor to its base-emitter voltage (V_{BE}) using the basic form of the Ebers-Moll equation:

$$I_E = I_S \exp\left(\frac{qV_{BE}}{kT}\right)$$

Here, I_S is the reverse saturation current of the base-emitter PN junction, which is a property of the individual transistor; q is the electronic charge (the charge on one electron, which is a physical constant); k is Boltzmann's constant (another physical constant); and T is the absolute temperature (in kelvin). The equation applies for relatively large emitter currents ($I_E >> I_S$).

This equation can be rearranged to give the base-emitter voltage:

$$V_{BE} = \frac{kT}{q}\ln\left(\frac{I_E}{I_S}\right)$$

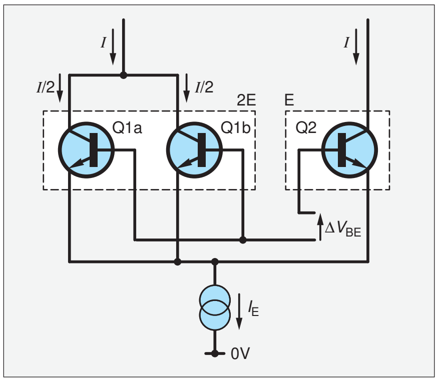


Fig.1. Concept circuit for obtaining ΔV_{BE} from two transistors with scaled sizes.

In which ln() is the natural logarithm function, the inverse of the exponential function. If we have two transistors with different emitter currents (I_{E1} and I_{E2}) and we assume the transistors are at the same temperature (T) and closely matched in terms of their physical characteristics (so $I_{S1} = I_{S2} = I_S$) then the difference between their base-emitter voltages (ΔV_{BE}) is:

$$\Delta V_{BE} = \frac{kT}{q}\ln\left(\frac{I_{E2}}{I_S}\right) - \frac{kT}{q}\ln\left(\frac{I_{E1}}{I_S}\right)$$

We are subtracting two logarithms, which is equivalent to dividing by the subtracted value inside the logarithm. So, we can write:

$$\Delta V_{BE} = \frac{kT}{q}\,ln\left(\frac{I_{E2}}{I_S}\Big/\frac{I_{E1}}{I_S}\right) = \frac{kT}{q}\,ln\left(\frac{I_{E2}}{I_{E1}}\right)$$

The transistor matching can be achieved with correct layout and fabrication of transistors on the same IC (it is much more difficult with discrete transistors). The result is the difference in V_{BE} only depends on temperature and the emitter current, not on the transistor characteristic (I_S). This is important because although two transistors can be matched on a single IC, different individual ICs will have transistors with different I_S. Furthermore, I_S varies significantly with temperature, resulting in complex and variable temperature dependence for transistors across multiple ICs. This means it is not possible to make an accurate temperature sensor based on a single transistor, but using the difference between two transistors on a single IC is a viable approach.

PTAT

If the two emitter currents are set to be in a fixed ratio (N), that is $I_{E2} = NI_{E1}$, then we get:

$$\Delta V_{BE} = \frac{kT}{q}\,ln(N)$$

In which ln(N), k and q are constant, so ΔV_{BE} is proportional to absolute temperature with a positive relationship (ΔV_{BE} increases with increasing temperature). This is referred to as a 'Voltage Proportional to Absolute Temperature', or V_{PTAT}:

$$\Delta V_{BE} = V_{PTAT}$$

If we have $N = 10$ (as used in the LM35) V_{PTAT} in the above equation is 198µV/°C , which can be scaled to give 10mV/°C or other convenient 'round number' values (eg, for the Fahrenheit scale).

Setting up two independent currents of ratio N (as might be implied by the above discussion) is not the most practical way to create a temperature sensor. It is easier to scale the effective size of the two transistors and ensure exactly the same current passes through both of them. Scaled transistors can be implemented on ICs using transistors with multiple emitters, which is equivalent to multiple parallel (ideally) identical transistors.

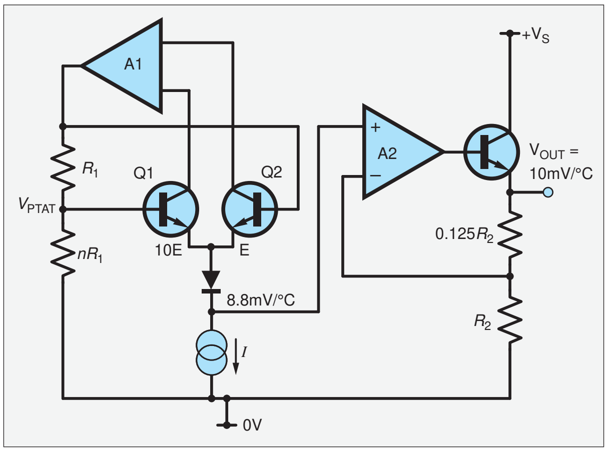


Fig.2. LM35 simplified device schematic (based on Texas Instruments data sheet).

Sensor circuits

An example of this principle is shown in Fig.1. Q1 has two emitters and is represented as two identical parallel devices (Q1a and Q1b) in Fig.1. Q2 is a single transistor identical to Q1a and Q1b. The circuit is set up so that the same current (I) flows through Q1 and Q2, which means that the currents in Q1a and Q1b are both $I/2$ (the currents split equally between Q1a and Q1b because they are identical). The total current is kept constant and well controlled by the current source I_E – the equality of currents in Q1 and Q2 is ensured by circuitry not shown in Fig.1.

The V_{BE} voltages of Q1a and Q1b must be equal due to their parallel connection and each transistor has a current of $I/2$ with respect to Q2. Thus, the difference in base-emitter voltages is given by the ΔV_{BE} equation above, with $N = 2$, and so has a linear dependence on temperature (PTAT).

The circuit in Fig.2 is a simplified schematic of the LM35. The core of the circuit, Q1 and Q2 and the current source, is configured as in Fig.1, expect that there is a ten-times rather than two-times relationship between the transistors (10E and E on Fig.2). The differential current-in, voltage-out amplifier A1 is in a feedback loop controlling the difference between the base-emitter voltages (ΔV_{BE}) so that the collector currents of Q1 and Q2 (A1's inputs) are forced to be equal.

The ΔV_{BE} voltage is across R_1, so it will cause a current of $\Delta V_{BE}/R_1$ to flow through it. Assuming this is much larger than Q1's base current (which can be ignored) the same current flows in the resistor nR_1, so the voltage across it will be $n\Delta V_{BE}$. As an end of nR_1 is grounded, the voltage at the base of Q1 will be a PTAT voltage related to the V_{BE} difference. The output of the LM35 is derived from the voltage at the current source, which is two diode voltage drops below Q1's base voltage (these also contribute to the final output temperature coefficient). This voltage is scaled by the voltage amplifier A2, whose gain is set to provide a 10mV/°C variation with temperature at the output.

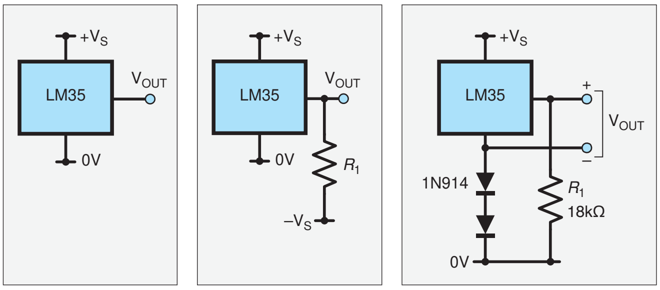
The circuit in Fig.2 can be calibrated after fabrication by adjusting nR_1 using various trimming techniques, for example, fuse links across the resistors in a series string can be blown to increase the resistance. The calibration sets the correct temperature coefficient of the output voltage.

For a more detailed discussion of the operation of these devices, including calibration and nonlinearity compensation (not covered here) refer to *LM34/LM35 Precision Monolithic Temperature Sensors, National Semiconductor Application Note 460*, October 1986.

The LM35 IC and variants

The LM35 is a three-terminal device which has two power pins and an output pin producing a voltage which, as described above, varies linearly with temperature at 10mV/°C. There are a large range of similar devices, some with similar part numbers such as those listed above. These devices vary in various ways, such as the temperature scale, supply voltage range, and presence, or otherwise, of a power-saving shutdown pin. Most devices are aimed at the Celsius scale, but the LM34 provides a 10mV/°F output.

Even with a given basic part number, such as 'LM35', there are quite a few variants with different packaging (such as TO92, TO220, TO46, SOT and SOIC), different accuracy ratings and different temperature ranges – not all devices provide the full datasheet headline range. This diversity means that care must be taken



Left to right (all based on Texas Instruments datasheet): Fig.3. Basic LM35 circuit; Fig.4. Full range LM35 circuit; and Fig.5. LM35 range extension on a single supply.

when ordering parts. Some variants are significantly more expensive than others.

Using the LM35 and similar devices

The most basic LM35 circuit is shown in Fig.3. This provides a 10 mV/°C output over the range 2°C to 150°C (20mV to 1.5V out) – notice that the lower end of the LM35's temperature range is not covered because the output of the LM35 cannot go negative in this circuit. The solution is to use a negative supply in the system and wire the LM35 as shown in Fig.4. The LM35 can only source current (current flows out, not in), but with the resistor (R_1) connected to a negative supply, a sourced current can result in an output voltage of either polarity with respect to ground. The value of R_1 is specified in the LM35 datasheet as $V_S/50\mu A$.

If a negative supply is not available, the circuit shown in Fig.5 can be used. The diodes raise the voltage at the ground pin of the LM35 above 0V (system ground) so the system ground is like a negative supply from the perspective of the LM35. The approach used in Fig.4 can then be applied. It may be tempting to assume the diodes drop about 0.6V or 0.7V (the typical assumption), but the LM35 may only be consuming 60µA, and at these current levels the 1N914 forward drop is in the range 0.45V to 0.5V. If we assume 0.9V *total*, then the rule for R_1 in Fig.4 gives the datasheet value for R_1 of 18kΩ.

The circuit in Fig.5 differs from the one in Fig.4 in that the output voltage is not referenced to ground. Various approaches to dealing with this are possible. Just the normal output pin of the LM35 could be used, on the assumption that the diode drop is constant, subtracting this to get the temperature voltage, but this is very likely to be inaccurate as a diode drop value will change (eg, with temperature and LM35 supply current). Better approaches are to use the differential output (as shown) with a differential input amplifier, a differential input ADC, or to measure both voltages separately (eg, using multiple ADC channels) and subtract in software.

The circuit in Fig.5 is not very convenient, so a better approach with a single supply is to use a different chip, which can measure negative temperatures on a single supply. Examples include the TMP36 from Analog Devices and the LM50 from Texas Instruments. These devices have a 10mV/°C output, like the LM35, but with a +0.5V offset (so output 0.5V at 0°C, 0.75V at 25°C and 0.25V at –25°C) which facilitates output of voltages representing negative temperatures. An example circuit for the TMP36 is shown in Fig.6. Note the shutdown pin which is available on some package options.

Long leads

It is not uncommon to need to locate a temperature sensor away from the main circuit board. This may make the sensor wiring susceptible to noise pick-up from various sources. Fortunately, in most situations, temperatures change relatively slowly and appropriate low-pass filtering can reduce noise without excessively impacting the measurement process. As usual, shielded and/or twisted pair cable should also be used for lengthy sensor connections to reduce the amount of noise pickup.

Another issue with long sensor connections, which should not be overlooked, is the capacitance of the cable. A common solution is to use an isolation resistor between the output and capacitive load. The capacitive drive capability varies significantly for different integrated analogue temperature sensors, but in general, the series resistor approach is applicable (see Fig.7) and sensor datasheets may provide advice on appropriate values (this is the case for the LM35 and TMP36). Alternatively, an *RC* damper may be used (see Fig.8). The series resistor, plus cable capacitance, or a damper circuit, also forms a low-pass filter assisting reduction noise.

As with many integrated circuits, it is often a good idea, or even essential, to connect one or more supply decoupling capacitors as close as possible to the sensor IC. A typical value is 0.1µF, but the datasheet should be consulted for the most appropriate values. The circuits in Fig.7 and 8 show typical circuits for using temperature sensors on long cables. In more extreme situations (very long cables, high electrical noise) converting the sensor voltage to a current for transmission down the cable may be a better approach.

Layout considerations

There is more to temperature sensing than the circuit design – the physical structure is also important to ensure that the right temperature is being measured and device operation is not undermined. The leads of the device conduct heat to the chip inside the package, so in situations where a device is glued to the surface of an item whose surface temperature is being measured, the air temperature surrounding the leads may have an unwanted influence on the reading – particularly for plastic package devices. The LM35 datasheet advises covering the leads with epoxy if this might be a problem (if the surface and air temperatures may be different). Metal package devices are less problematical in this respect and can be soldered to metal surfaces being measured, but only if the metal to which it is attached can be at the device negative supply lead voltage.

If the device is used to measure liquid temperature, then arrangements must be made to keep it and its wiring dry and electrically insulated from the liquid. Moisture may also be an issue where cold temperatures are being measured and condensation may occur on the device or surrounding circuitry – suitable coatings can help in these situations.

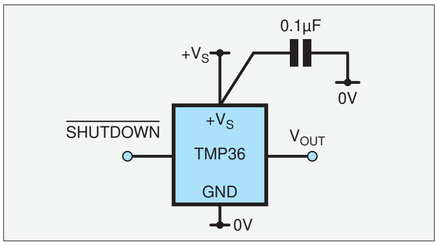


Fig.6. TMP36 circuit with –40°C to +125°C range (based on Analog Devices datasheet).

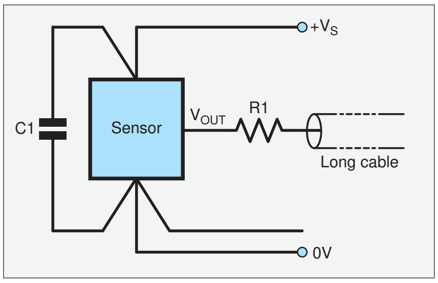


Fig.7. Analogue integrated temperature sensor on long cable with decoupling capacitor C_1 and load capacitance isolation resistor R_1.

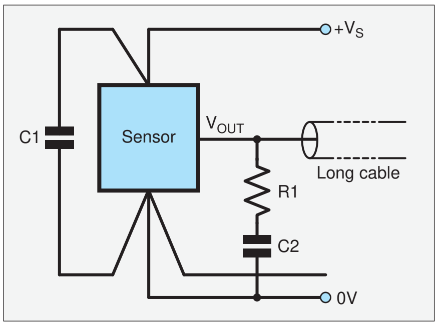
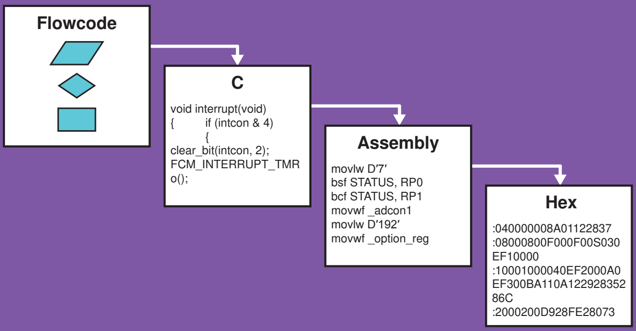


Fig.8. Analogue integrated temperature sensor on long cable with decoupling capacitor C_1 and *RC* damper (R_1 and C_2).

Flowcode
Graphical
Programming



```
void interrupt(void)
{         if (intcon & 4)
          {
clear_bit(intcon, 2);
FCM_INTERRUPT_TMR
o();
```

```
movlw D'7'
bsf STATUS, RP0
bcf STATUS, RP1
movwf _adcon1
movlw D'192'
movwf _option_reg
```

```
:040000008A01122837
:08000800F000F00S030
EF10000
:10001000040EF2000A0
EF300BA110A122928352
86C
:2000200D928FE28073
```

Programming with Flowcode – Part 3: LCD temperature monitor

This month, we will expand our use of Flowcode by showing how to use a PIC16F88 and RedBoard/Uno to measure the output of an analogue sensor and display it on a 16x2 LCD alphanumeric display. The sensor we have chosen measures temperature, but similar principles would apply to analogue voltages from other sources.

Do note that we are following on from *Parts 1* and *2* in the January and February 2022 issues of *PE*. To avoid a lot of tedious repetition we will not rehash the ideas already covered, so you do need to have read those articles to get the most out of this project. (We will, however, provide some handy links to refresh your memory.)

Last, but not least, all the components used are in the 'free' section of Flowcode – no purchase needed! (See the box at the end of this article for further details.)

Sensor and display choices

Flowcode libraries include different types of temperature sensors which fall into two distinct categories: analogue and digital. For analogue sensors, the output voltage is proportional to the measured temperature, and it interfaces to a single microcontroller ADC pin. (Note that what I'm discussing here are not sensors like thermocouples or thermistors, but their IC equivalents; for example, the well-known LM35.) The output of digital sensors is either 0, 5V or 3.3V depending on the sensor and microcontroller used.

In this project we take the analogue option via the popular LM35 sensor from Texas Instruments, but the free Flowcode component we use can support all of the following temperature sensor ICs: AD22100, AD22103, LM135, LM235, LM335, LM35, LM60, MCP9700, MCP9701, TC1047, TMP35, TMP36, TMP37 and TSIC301. These devices provide varying degrees of accuracy, linearity and range – you choose the one best suited to your application and budget. (See this month's *Circuit Surgery* for more insight into the LM35.)

For the 16x2 (16 columns, 2 rows) display, we will use an LCD based on the Hitachi HD44780 parallel interface LCD controller chip. The display connection pins used will be the same regardless of the display size you use. For example, a 20x4 LCD uses the same pins as a 16x2 device. Fig.1 shows a typical 16x2 LCD.

These LCDs operates in two modes: eight-bit and four-bit. The number of bits determines how many data pins are used. We will use four-bit mode via the four data pins D4 to D8, (eight-bit mode would use all eight data pins, D0 to D7).

Project overview

We are going to create a flowchart that reads the voltage on the microcontroller analogue input pin which is connected to the sensor, and then on the LCD, show the Celsius temperature and one of three messages: 'Too cold', 'Just right' or 'Too hot'.

Also, three microcontroller pins (A0, A1 and A2) are assigned to message indicator LEDs (via 330Ω current-limiting resistors).

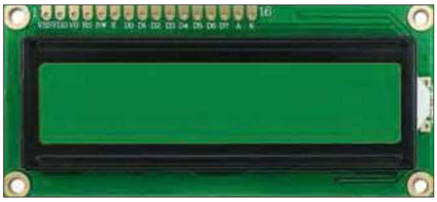


Fig.1. Typical 16x2 LCD.

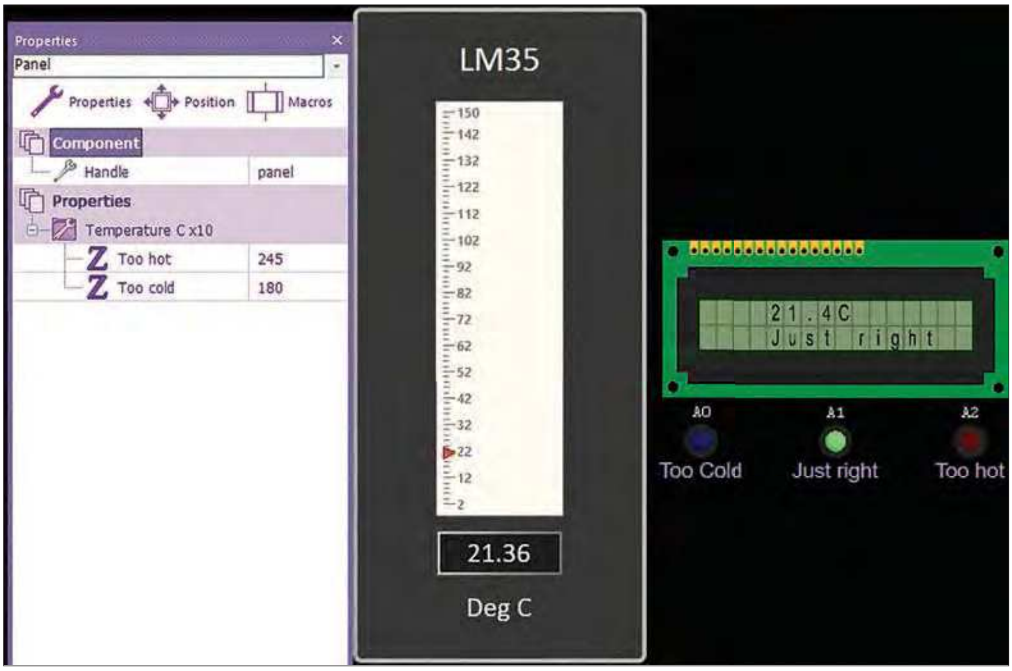
Fig.2 shows what the Flowcode 2D Panel will look like when the project is completed and running in simulation mode.

Designing the flowchart

Let's get started on the Flowcode design. (Again, remember that I am assuming you have read *Parts 1* and *2*.)

First, run Flowcode and select New Project, the last target device used should be in the new Embedded Project window. If it is not the one you want (PIC16F88) then right-click any device in the Choose a Target section and search for your device – see *First Steps* in *PE*, January 2022, p.39.

If you are using a RedBoard/Uno, make sure the correct com port is selected (see *PE*, January 2022, p.41-42).



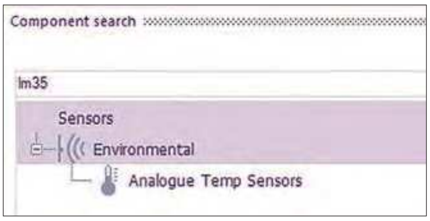
Panel

Handle	panel
Too hot	245
Too cold	180

21.36
Deg C

21.4C
Just right

Fig.2. Simulation of the LM35-based temperature display running on the Flowcode 2D Panel.


Component search

lm35

Fig.3. Searching for an LM35 will bring up an 'Analogue Temp Sensors' device.

Whichever microcontroller you choose, ensure it is set with the same parameters, for example clock speed, as used in *Part 2*.

Components

The first step in creating a project is adding the required components. The easiest way is to select the Components Libraries Ribbon and then use the search tool. We need the following four components:

Analogue temperature sensor – search for one of the free temperature sensor ICs listed above, for example 'LM35' or 'TMP36'. Note that when you have found your device it will appear with the label `Analogue Temp Sensors` (see Fig.3).

Display – Search for 'LCD'. To keep this cost-free, scroll down to `LCD (Generic, 16x2)` for a two-row device, or for four rows, use `Combo (BL0114)`. If you choose the 16x2 device, then the component name that appears in the 2D Panel is `lcd_16x2`.

LEDs – Search for 'LED'. The free version is `LED (5mm, Panel)`; we will need three.

Static text – search for 'Static', add one component, resize and then copy and paste twice from the 2D Panel. To edit text, right-click and select properties and then change the default 'sample text' to the text you want.

When the correct component is found, right-select it and add it to the 2D dashboard panel. If you need multiple components of the same size, it's quicker to add one component, adjust its size, and then copy and paste it on the 2D Panel.

Likewise, if you have multiple components you would like to resize by the same percentage, hold the shift key and select the components you want to process. Now select the resize icon.

Arrange your components so that the 2D dashboard panel looks like Fig.2. I drag the components to roughly the correct position and then fine tune them using 'Position' from Properties (see *PE*, January 2022, p.42)

Next, change the LEDs to the correct colour and rename them (see *PE*, January 2022, p.43) to match Fig.2.

Make sure the connections are correct for all the components (see Table 1). The numbers in brackets are the pin connections of the LCD. For example, the connection for Data 0 goes from pin B0 of the microcontroller to pin 11 on the LCD.

Variables

As well as components, we need to create the variables required for the project, and for good practice initialise them, even though not all of them actually require it. The variables used are shown in Table 2.

Refer back to February 2022, p.38-39 for details on how to set up variables. There are three kinds of variable here:

Byte – these are 8-bit, unsigned numerical values that can have a value from 0 to 255.

Int – these are signed numerical values that can have a value from –32678 to 32677.

String – these contain not just numbers, but also other characters, often letters from the ASCII code and are typically used for short pieces of text

Component	PIC16F88	RedBoard/Uno
LEDs		
Too cold	A0	D5 = Digital 5
Just right	A1	D6 = Digital 6
Too hot	A2	D7 = Digital 7
Temp Sensor	A4	C4 = Ana 4
LCD		
Data 0 (11)	B0	B0 = Digital 8
Data 1 (12)	B1	B1 = Digital 9
Data 2 (13)	B2	B2 = Digital 10
Data 3 (14)	B3	B3 = Digital 11
Register Select (4)	B4	B4 = Digital 12
Enable (6)	B5	B5 = Digital 13

Table 1. Component-PIC/RedBoard connections.

Variable Name	Type	Initilise Value
Offset	Byte	0
State	Byte	0
TempInt	Int	0
TempString	String	[Array length = 5]

Table 2. Flowcode project variables

In our temperature display project, we are going to use four variables:

Offset – used in Calculation boxes to set the level of switching hysteresis.

State – result from the Calculation box that is used by the Switch.

TempInt – represents the temperature value after x10 processing (see below).

TempString – text message displayed on the LCD.

When you declare a string variable you need to define the maximum number of characters used, in our case this is 10. As detailed last month, click Global Variables in Project Explorer and when the pop up window appears, choose 'String'. Name the string `TempString`, and in array length enter `5`, then select Add Array Dimension. You should now see `TempString[5]` for the string name.

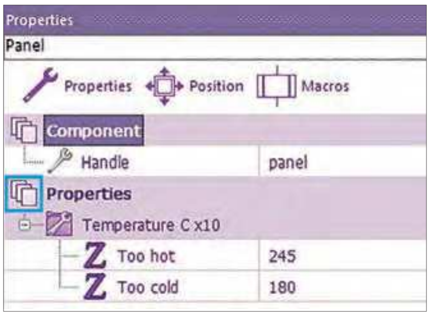
Last, we need to assign values to a pair of Properties. These are the low and high temperature settings, which are used to trigger the LEDs and messages. We do this by adding these variables to the Properties of the 2D Panel.

To do this, just select Properties from a blank area of the 2D panel or select Properties from the View ribbon. Next, select the drop-down outlined in blue in Fig.4, and click on New category. Now enter a suitable name – `Temperature C x10`. Select the spanner folder, click on Add new and make the following settings:

Cosmetic name – type in a name (you can use spaces); for example, `Too hot`.

Property type – choose Signed integer.

Property variable – this is the variable's name, but note that that spaces or starting with a number are not allowed; for example, use `TooHot`.


Properties

Panel

Handle panel

Temperature C x10

Too hot 245

Too cold 180

Fig.4. Adding the low and high temperature settings variables.

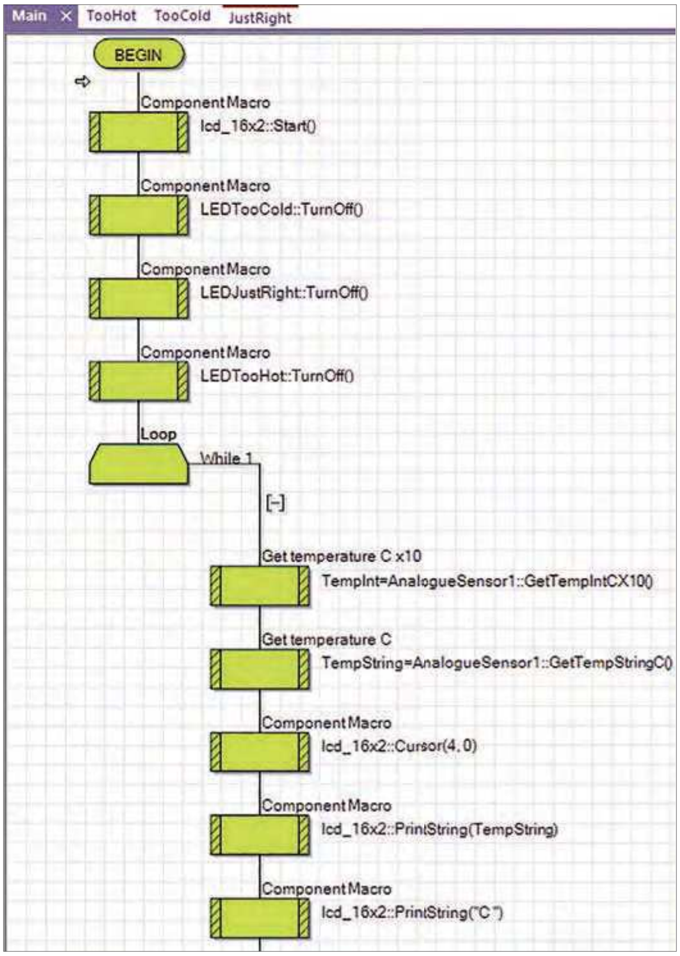


Fig.5. Building the first part of the flowchart.

Now you can set the high-temperature limit value, which is 245 for `TooHot`.

Repeat this process for a temperature setting called `Too cold` and use `TooCold` as the variable name, with value `180`. You should now have created what is shown in Fig.4.

Software development

Before we get into the nitty-gritty of flowchart building, let's define what we want the software to do. There are eight main components to the software:

1. Begin and End, which are common to all programs, and are built into the Main Macro
2. Initialise the LCD
3. Set the LED initial conditions
4. Start an endless loop, inside which the following three steps are performed
5. Measure temperature
6. Write the temperature to the LCD
7. Compare temperature with set parameters, then turn on one LED according to the temperature and write one of three messages to the LCD
8. Return to start of loop.

Let's run through each of these in turn and build up our temperature display program.

1. Begin and End

Remember, in Flowcode, the whole program lives within a Macro called 'Main' (see top of Fig.5). If you examine Main immediately after creating a new project, it will just contain the Begin and End icons. All the components we need to add are placed after Begin.

2. Initialise the LCD

All components in Flowcode that have initial parameters must be initialised at the start of the program – an LCD is such a component. Therefore, the first component function to be added initialises the LCD. We do this with the Component Macro `lcd_16x2 Start`. You can find it by looking at the display component's list of Component Macros – click the '+' in the little square to the left of the `lcd_16x2` component in Project Explorer (Fig.6). Now, from the bottom of the list, drag the `Start` Component Macro to underneath Begin in Main.

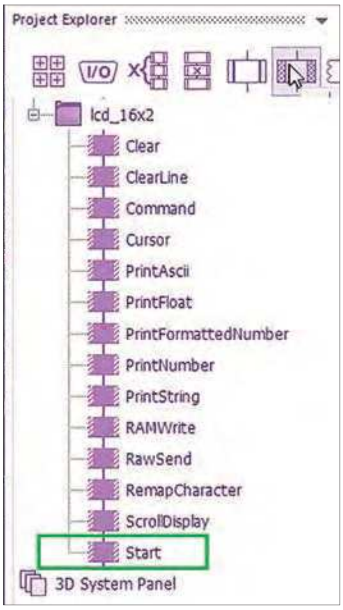


Fig.6. Locating `lcd_16x2 Start`, the LCD's initialising Component Macro.

3. Set the LED initial conditions

LEDs do not need to be initialised to operate properly, but for the project to work correctly we do need to set their initial state. The initial state for the three LEDs is off.

4. Start an endless loop

4. Next, drag a Loop from the Command icons and place it below the LEDs' initial state functions.

5. and 6. Measure temperature and write to LCD

Now we turn to reading the sensor in degrees Celsius and writing it to the LCD. The next stages involve more Component Macros. These are found in a similar way to `lcd_16x2 Start` described above. The following Component Macros are dragged inside the main loop in the following order, top to bottom:

`AnalogueSensor1 GetTempIntCX10` – multiplies the temperature value by ten and assigns the result to a variable called `TempInt`. (See below for an explanation of why this is done.)

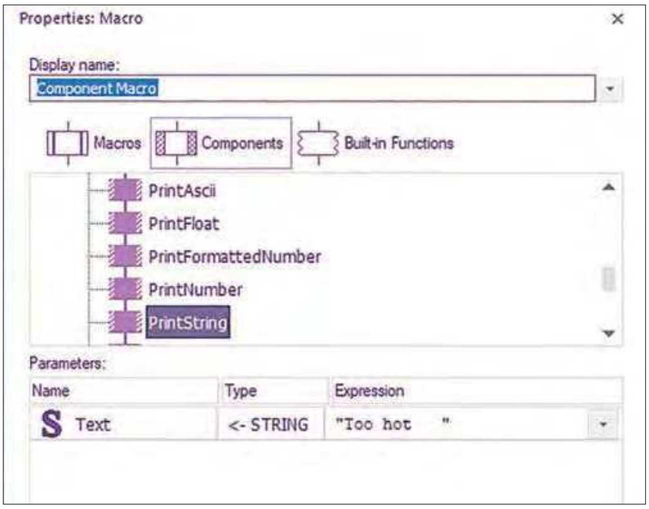
`AnalogueSensor1 GetTempStringC` – assigns the temperature value as a string (eg, '24.8') to a variable called `TempString` which is used by the LCD to display the temperature.

`lcd_16x2 Cursor x,y` – defines the cursor starting position: `x` = column, `y` = row. We use `lcd_16x2 Cursor 4,0` – note the x position values start at 0, so for 4,0 the cursor will start at the fifth (not fourth) location from the left, and in the first row. Refer back to Fig.2 – this is the location of the number '2' in '21.8', thus helping to centre the displayed value.

`lcd_16x2 PrintString` – uses the `TempString` variable to print the temperature value to the display.

`lcd_16x2 PrintString` – uses the same Component Macro again to print 'C ' Note the single space after the 'C', which is used to ensure all of the previous temperature value characters are overwritten.

The reason behind the need for the times-ten Component Macro `AnalogueSensor1 GetTempIntCX10` is that it allows us to convert a floating-point variable like 24.5 into an integer version without a decimal point; ie, 245. Unfortunately, floating variables with decimal point takes up a lot of memory and for that reason, we are unable to use them on the PIC16F88. Using an integer version allows us to get round this problem.



Display name: Component Macro

Parameters:

Name	Type	Expression
Text	<- STRING	"Too hot "

Fig.7. Add text to `PrintString` using the Expression box.

We need to add several different pieces of text to the `lcd_16x2 PrintString` Component Macros. To do this, go to the Components listing in Project Explorer, scroll down to `lcd_16x2`, open the component list and drag `lcd_16x2 PrintString` to where it will be used. When you release the Component, a Properties window will open. In the Expression section of the pop up window, enter the required text inside double quote marks; ie, `"Too hot    "`, see Fig.7.

Note – there's one exception to the double quote mark rule. When adding the `lcd_16x2 PrintString` Component Macro for `TempString` in Fig.5, do not use double quotes. Instead, use the Expression drop-down and select `TempString`.

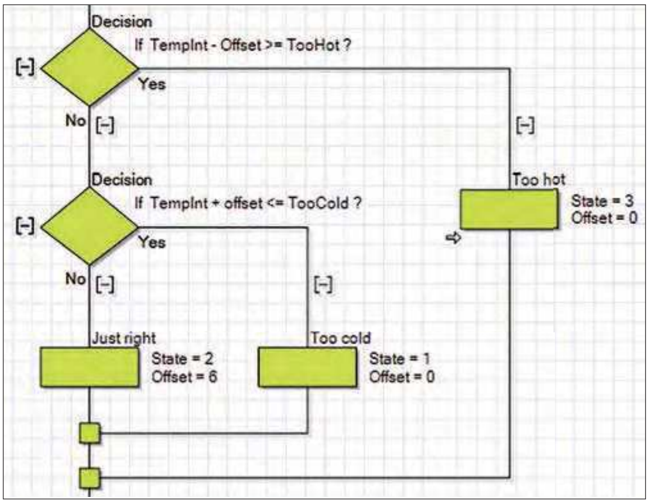


Fig.8. Building the Decision/Calculation branches.

Note: In Fig.5 I edited the loop Component Macro labels (above-right of the green boxes) in the second and third components. For example, I used `Get temperature C x10` to make the flowchart easier to read. It is good practice to add little reminders and annotations as you proceed with a flowchart.

When you have added all these items check that your flowchart matches Fig.5.

7. LEDs and messages

The next step is a bit more complex, here we will set up the Decision branches that select which LED to illuminate and which message to send to the LCD. For the following description, refer to Fig.8, which follows on directly after the bottom of Fig.5.

There are two kinds of operation in Fig.8. First, the Decision branches. The program asks if the temperature is too hot; if it is, then turn on the 'Too hot' LED. If No, then the next Decision branch asks if the temperature is too cold – if yes, then turn on the 'Too cold' LED. If No, then turn on the 'Just right' LED.

There is a subtle problem with this simple approach to switching LEDs on/off based on a temperature threshold. If the temperature lies on the border between two states (eg, 'Just right' and 'Too hot') then the output of the PIC's ADC may oscillate between two very closely matched values and cause the LEDs to blink. This is not really a serious problem if all you are doing is switching LEDs, but if you were switching relays or larger equipment it could be a real problem to repeatedly switch devices on and off in rapid succession. The answer to this kind of problem is to add some hysteresis. With this in mind, our Decision branches use the following two expression:

```
TempInt – Offset >= TooHot
TempInt + offset <= TooCold
```

It is the `Offset` that adds the crucial hysteresis, which is used to suppress potential oscillation.

When the temperature is rising, from Just right to Too hot, Too hot must be 0.6 degrees hotter than the 245 (ie, 251) Too hot setting to trigger the Too hot LED/message.

When the temperature is falling from Too hot to Just Right, the trigger is 'just under' 245.

Likewise, when the temperature is falling from Just Right to Too cold, the trigger point is 0.6 degrees colder than the 180 Too cold setting (ie, 174).

When the temperature is rising from Too cold to Just Right, the trigger is 'just over' 180.

To add the items in Fig.8. First drag a Decision (from the Command Icon section, see *PE*, February 2022, p.40) to underneath the lower `lcd_16x2 PrintString` Component Macro (in Fig.5). Next, add the second Decision, but ensure it is above the first Decision's 'return square' – ie, it is nested inside the first added Decision. Double click the Decisions and add the hysteresis expressions.

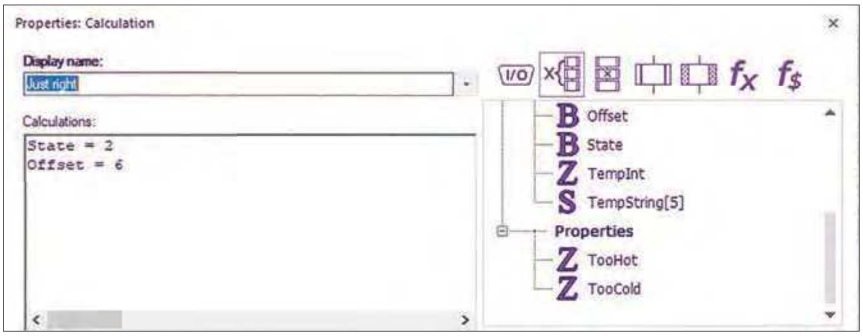
Next, add the (rectangular) Calculation boxes (from the Commands Icon section). Double click on each one and the Calculation window will pop up. Drag the two variables `State` and `Offset` from the right-hand side of the window under Global Variables to the blank left-hand side of the pop up (see Fig.9).

Next, edit `Offset` so that it says `Offset=0` (or `Offset=6`). Likewise, edit `State` so that is says `State=1` (or `2`, or `3`). You should now have created the contents of Fig.8.

Calculation Component Macros

So what do the three Calculation boxes mentioned above do? Their purpose is not to directly turn on or off LEDs, or send messages to the LCD, What they do is set variable values that are picked up by the next item in the flowchart – a Switch component – which contains User Macros that do control the LEDs and LCD messaging.

Now move on to Fig.10, which is placed immediately after the decision branches section. It uses the `State` variable from the Calculation boxes and there are three possible outcomes depending on the value of `State`.



Display name: Just right

Calculations:

```
State = 2
Offset = 6
```

Fig.9. Adding `Offset` and `State` in the Calculation Properties window.

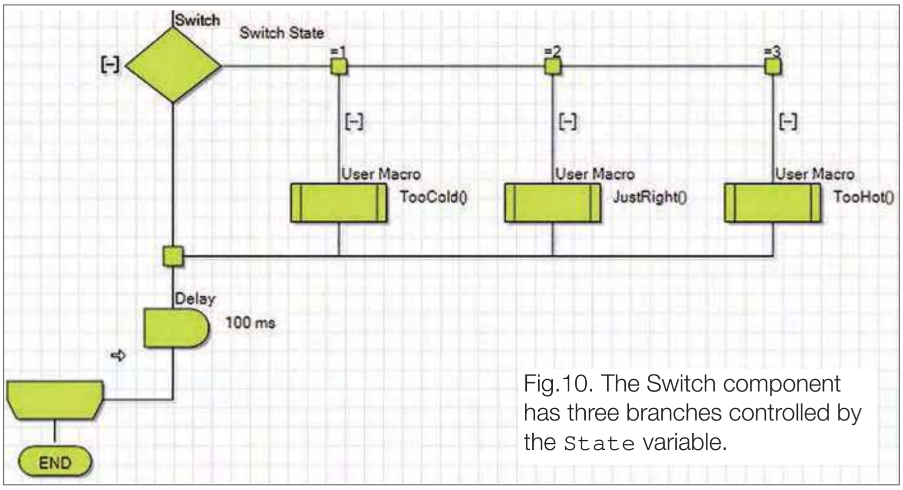


Fig.10. The Switch component has three branches controlled by the `State` variable.

The Calculation boxes can generate one of three `State` values: 1, 2 or 3 in response to the Decision results. For example, if the first Decision answer is 'Yes' to the question, 'is the temperature too hot?', then `State` = 3. Likewise, if the second Decision answer is 'Yes' to the question, 'is the temperature too cold?', then `State` = 1.

The way Switch works is that when it reads the variable `State`, the only Switch branch accessed by the program is the one that matches the value of `State`. This accessed branch contains a User Macro that then lights the appropriate LED and sends the correct message to the LCD.

To build the section shown in Fig.10 we must first create our User Macros. Start by adding three User Macros, as shown in Fig.11. (We covered User Macros in *PE*, February 2022, p.38). Next, to assemble the Macros, copy the flowcharts in Fig.12, paying attention to the settings. (LEDs were covered in *PE*, February 2022, p.38.)

Do pay attention to adding the spaces in the `lcd_16x2 PrintString` Component Macros. The use of spaces is particularly important to get the LCD to display text messages properly. For example, if the display shows 'Just right', but then changes to 'Too hot'. Without spaces in the `TooHot` user macro, the display will show 'Too hotht' because the 'ht' of 'Just right' is not overwritten with spaces.

Now we can add the User Macros to the Switch. Double click the Switch and the Properties window will appear. From the Switch dropdown, select Global Variable `State`. Next tick the first three 'Cases' and the three `State` values will appear by default. Click OK.

Next, drag your three User Macros to the appropriate Switch branch, determined by the `State` value at the top of the branch. You should now have what is show in Fig.10.

The last component to add is a delay of 100ms at the end of the main loop (but still inside the loop). This stops the display from updating too quickly.

Download the complete flowchart

The complete flowchart is too large to print on a page, but you can download the PIC and RedBoard/Uno Flowcode files from the April 2022 page of the *PE* website.

Hardware Setup – PIC16F88

Although the PIC16F88 (like most microcontrollers) can run off its internal oscillator, it does cause some Flowcode complications, so this will be covered in a future article. For now, we'll stick with the standard external crystal/capacitors oscillator we used in *PE*, February 2022, p.41 (see Fig.27). We will need the following parts (one each, unless indicated).

Parts list – PIC16F88 circuit

PIC16F88
LCD 1602 (or 2004) with HD44780 chipset

LED	red
LED	green
LED	blue
Crystal	196608MHz
Resistor	330Ω x3
Capacitor	22pF x2
Capacitor	100nF
Trimmer	10kΩ

Solderless breadboard

Assembly

Construction is simple – just follow the diagram in Fig.13. Do ensure the supply wires to the microcontroller and the V_O connection from the temperature sensor are as short as possible and use the thickest-gauge wire that can plug into the breadboard. Any cable resistance introduced will affect the readings. (Note that if the temperature reading is out because of the leads, there is temperature compensation within the sensor properties.) When built, you should have something like Fig.14.

Hardware Setup RedBoard/Uno

Building the RedBoard/Uno hardware is just as straightforward – in fact, it's simpler because there are fewer components – simply follow the diagram in Fig.15.

Parts list – RedBoard/Uno circuit

LCD 1602 (or 2004) HD44780 chipset

LED	red
LED	green
LED	blue
Resistor	330Ω x3
Trimmer	10kΩ

Hook-up wire
Solderless breadboard

Loading the software

To test the Flowcode simulation, and then load the software in the PIC or RedBoard, follow the same sequence of operations explained before – see p.40

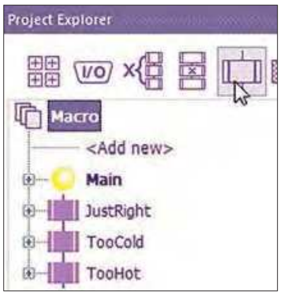


Fig.11. Adding three User Macros in Project Explorer.

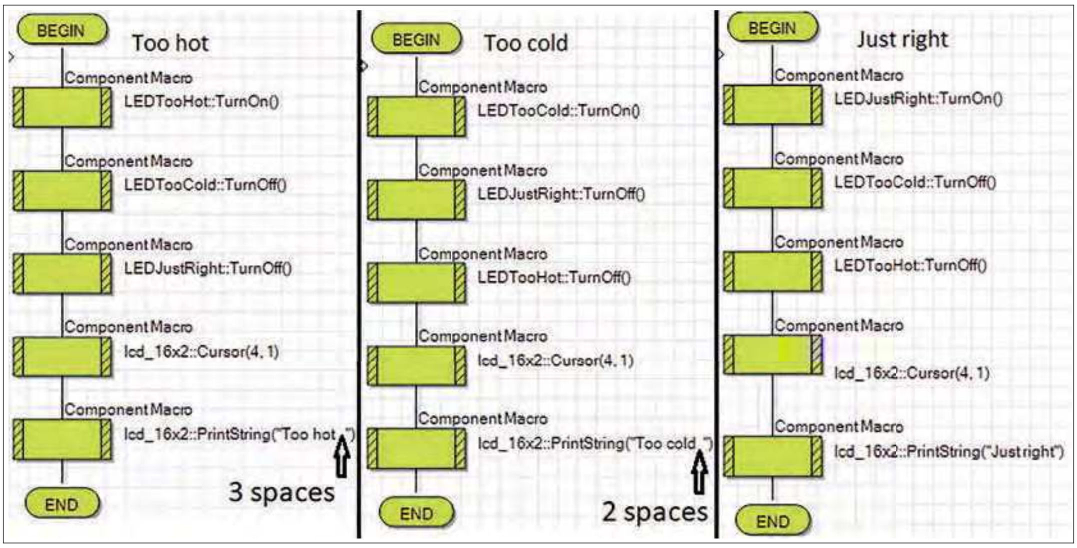


Fig.12. Layout of the three User Macros.

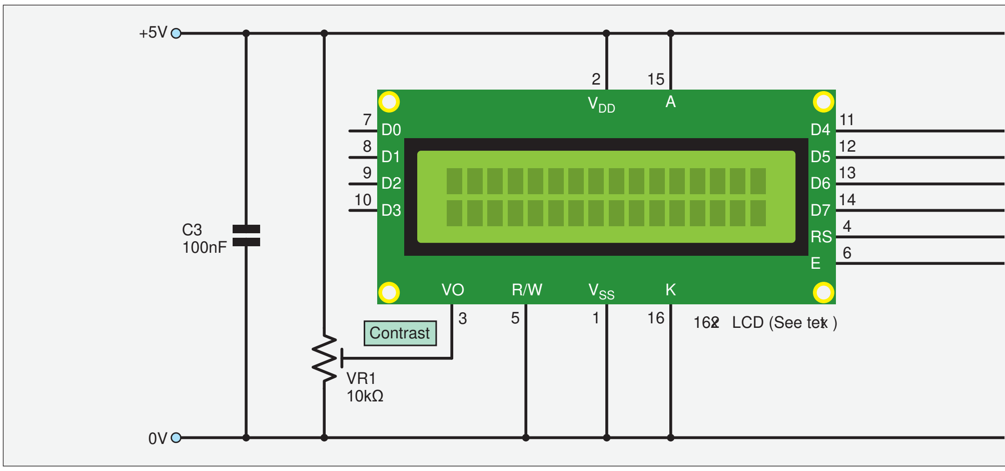


Fig.13. Circuit diagram of the PIC16F88-based temperature display.

for simulation and p.42 for programming the hardware in *PE*, February 2022.

Running the circuit

Once you have programmed your hardware, simply power up and the circuit will start automatically.

Basic fault finding

This is a simple circuit, but there's always room for gremlins! Here are a few tips to getting your circuit running smoothly.

If the display is blank, then it could be a contrast issue. The trimmer supplies a voltage to pin 3 of the LCD. The ideal pin 3 voltage for my setup is around 0.5V.

If the LCD's top row is fill with squares, then this is often a sign that the LCD has not being initialised. That could be down to wrong connections or an open circuit. Alternatively, the microcontroller might not be running at the correct speed or has the wrong code installed.

A double row of filled squares indicates either the wrong code within the microcontroller, too much contrast or poor connections.

If the wrong temperature is displayed, then either the sensor is connected to the wrong pin or the wrong component has been selected in Flowcode and an unexpected voltage level is present on

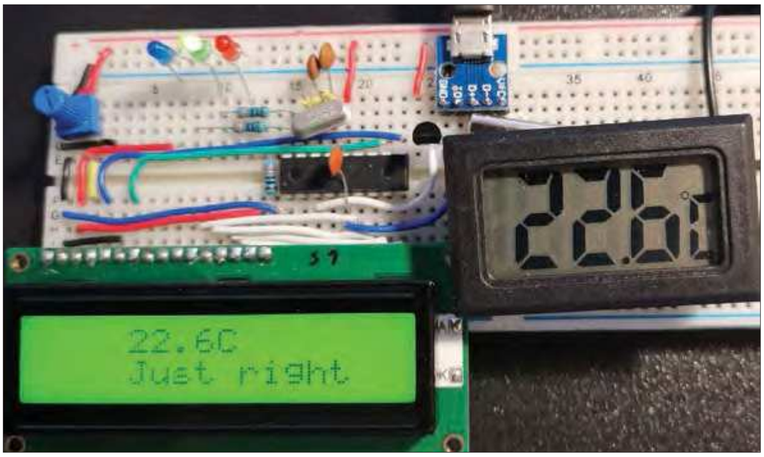


Fig.14. PIC16F88-based temperature display hardware.

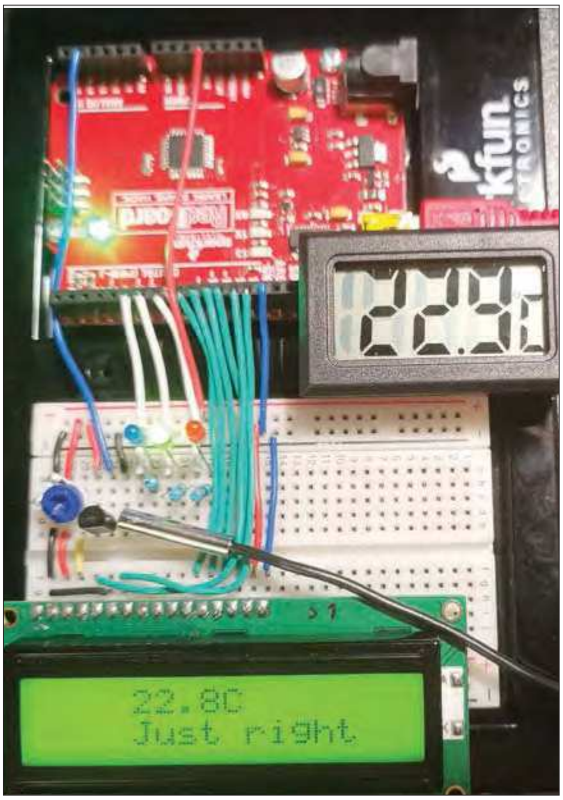


Fig.16. RedBoard/Uno temperature display hardware.

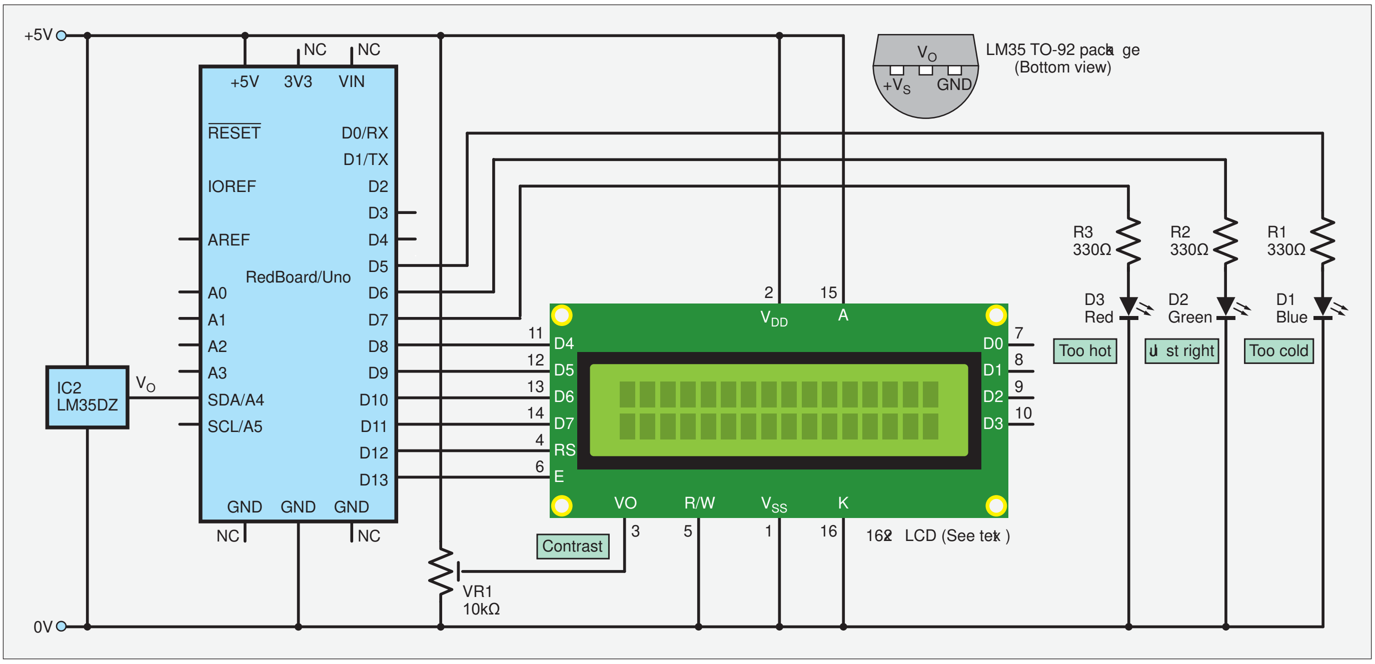
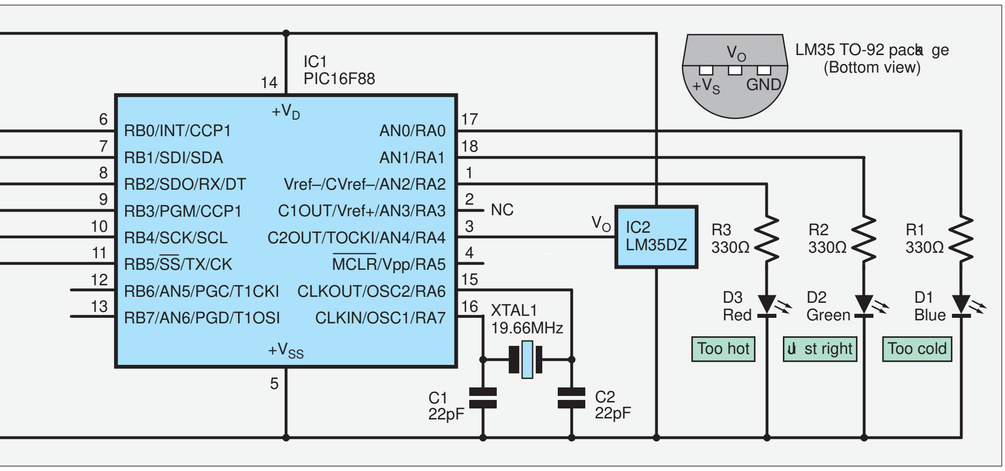


Fig.15. Circuit diagram of the RedBoard/Uno-based temperature display.



the microcontroller's input ADC pin. A performance guide for the LM35 is the output from the sensor is one hundredth of the temperature in Celsius. For example, at 21.3°C the output voltage is approximately 0.213V.

Still having problems? If you need help, then create a post at **https://flowcode.co.uk/forums/** and someone will help get your project up and running.

Martin Whitlock is an Applications Engineer at Matrix TSL – the company behind Flowcode.

Try Flowcode for *free!*

We hope you found this article interesting. If you'd like to try Flowcode for free then just go to **https://flowcode.co.uk/download/** and download the code. You'll get a 30-day free trial of the full version – but that's not all. Even after the 30 days are up your copy of Flowcode will continue to work, but at a reduced level with a limit on the size of program you can run and access to a more basic set of parts. However, for beginners it is still an ideal platform with which you can build and run programs, on for example, the Arduino Uno/RedBoard or a PIC 16F88 that we are using in this project.

Only when you are sure that you want to use Flowcode do you need to buy *inexpensive* access to say a Raspberry Pi or Bluetooth module. (See: **https://flowcode.co.uk/buy/process/** for all the modules available and what they contain.) What's more, as soon as you buy *any* module, the restrictions on the size of your code are removed.

If you get stuck with anything relating to Flowcode use (installation, software, creating flowcharts, compiling to hardware, hardware not working…) then there are forums set up to help you: **https://www.flowcode.co.uk/forums**

PE discount!
One more thing – Flowcode is deliberately designed to be inexpensive, but *PE* readers can get a further 20% discount when they use the code **PE20** at checkout.

Teach-In 8 CD-ROM
Exploring the Arduino

This CD-ROM version of the exciting and popular *Teach-In 8* series has been designed for electronics enthusiasts who want to get to grips with the inexpensive, immensely popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. *Teach-In 8* provides a one-stop source of ideas and practical information.

The Arduino offers a remarkably effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle wirelessly or via the Internet. *Teach-In 8* is based around a series of practical projects with plenty of information for customisation. The projects can be combined together in many different ways in order to build more complex systems that can be used to solve a wide variety of home automation and environmental monitoring problems. The series includes topics such as RF technology, wireless networking and remote web access.

PLUS: PICs and the PICkit 3 – A beginners guide
The CD-ROM also includes a bonus – an extra 12-part series based around the popular PIC microcontroller, explaining how to build PIC-based systems.

SOFTWARE
The CD-ROM contains the software for both the *Teach-In 8* and PICkit 3 series.

PRICE
£8.99
Includes P&P to UK if ordered direct from us

ORDER YOUR COPY TODAY!
JUST CALL 01202 880299 OR VISIT www.epemag.com

AUDIO OUT

By Jake Rothman

Analogue Vocoder – Part 6: Universal Audio PSU

Fig.1. Why does so much high-quality audio equipment have cheap and nasty £5.00 'wall-wart' power supplies? Our dog quickly dispatches them to landfill.

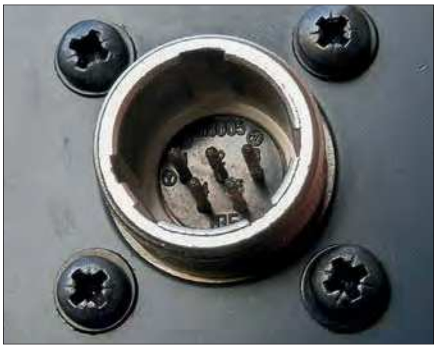


Fig.3. Power supply connectors are non-standardised. This mixer uses an unusual connector – the plug took some searching, and as usual, eBay failed, but Mouser delivered.

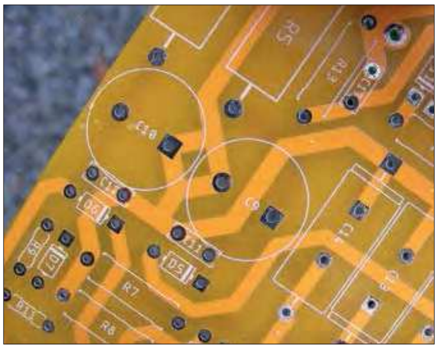


Fig.4. Careful track routing ensures low hum spikes. Here, PCB designer Mike Grindle has used special star earthing to separate the rectifier ground, quiet power ground and reference 0V.

Audio lives or dies by its power supplies, but all too often it runs off Lo-Fi plastic 'wall warts' that only last 13 months (ie, just long enough to get past the guarantee). In our house, some even get eaten by the dog! – see Fig.1.

Audio is based on transducers, amplifiers, signal processors and oscillators – running off power supplies. All too often the humble power supply is just an afterthought, so I reckon we're due a high-quality *Audio Out* power supply. Here's a big one especially designed for the *Analogue Vocoder* and microphone pre-amp, but it's also suitable for mixers, synthesisers and other systems with upwards of 40 op amps, where a higher current than normal (>250mA) is needed.

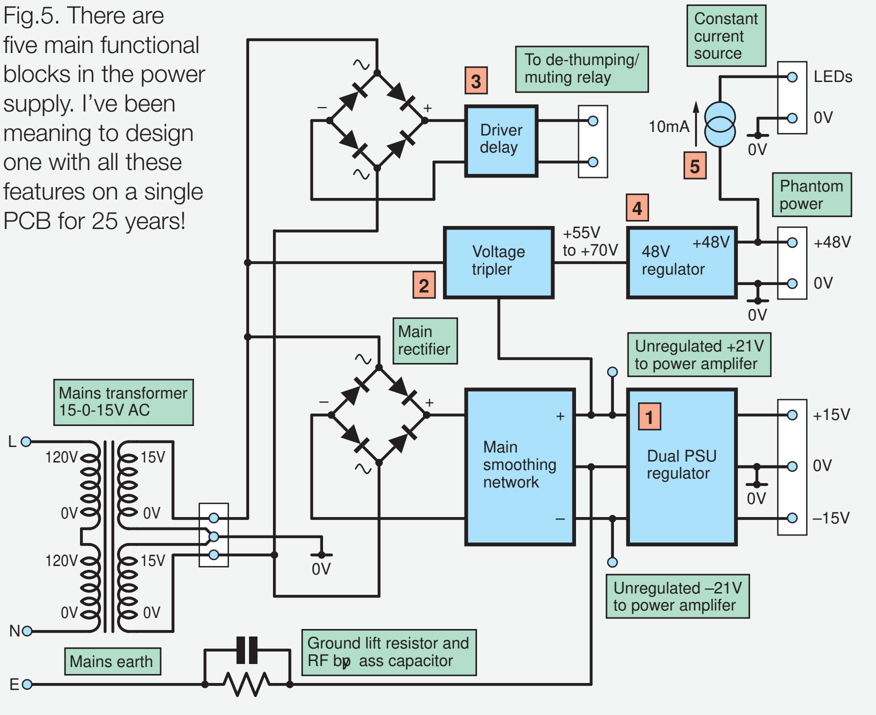
Another common requirement is a +48V rail for phantom-powered microphones.

Naturally, we'll also need the lowest possible noise, and don't forget hum is the main priority for audio power supplies.

This audio power supply provides ±15V dual rails at up to 750mA and +48V at 80mA. There is also a constant-current supply for front panel LEDs and a de-thump relay driver. The un-regulated ±21V power rails are also available for use with small power amplifiers driving speakers.

Fig.2. This excellent mixing desk was rescued, but minus its power supply.

Fig.5. There are five main functional blocks in the power supply. I've been meaning to design one with all these features on a single PCB for 25 years!



This design started when I found a mixer (Fig.2) in a skip. I never found the power supply, which uses a non-standard multi-pin power supply connector (Fig.3). (Remember to always use male/pins for inputs and female/sockets for outputs.) I was shocked when I saw such supplies on eBay were over £200 (second hand), so it made sense to design and build one.

Phantom generation

The BBC used to use a specially made toroidal transformer with a separate winding for the 48V section supplied by Canford Audio. I avoid expensive bespoke parts, so I generate sufficient voltage using a circuit driven by a standard 15-0-15V transformer.

Noise

The main source of electrical noise in power supplies is hum caused by capacitor charging currents from the rectifier, which induce voltages into the ground and power rail connections. This can easily be avoided if the layout is 'starred' correctly by feeding the rectifier outputs straight into the smoothing capacitor pins and then taking the power rail directly from that pin. It is essential to avoid sharing conductors for charging and power since the finite resistance of the PCB track will lead to charging pulses being superimposed on the output. (My PCB designer) Mike Grindle has taken great care in the PCB track layout to optimise this. Note the diamond track layout around the main smoothing capacitors (C7 and C8) shown in Fig.4. High-current tracks are also made as wide as possible.

It is not often appreciated that output decoupling capacitors can paradoxically increase regulator noise due to resonance with the regulators' output impedance. Regulators, as in all negative feedback amplifier systems, have an output impedance that rises with frequency; effectively it is 'an inductance'. This is a result of the open-loop gain declining with frequency. In the case of the LM3XX series of adjustable regulators this is in the order of 40µH. Ringing is also exacerbated by snubber capacitors used to suppress diode switching. Ironically, the use of high-quality low-loss

Fig.6. Power supply circuit diagram.

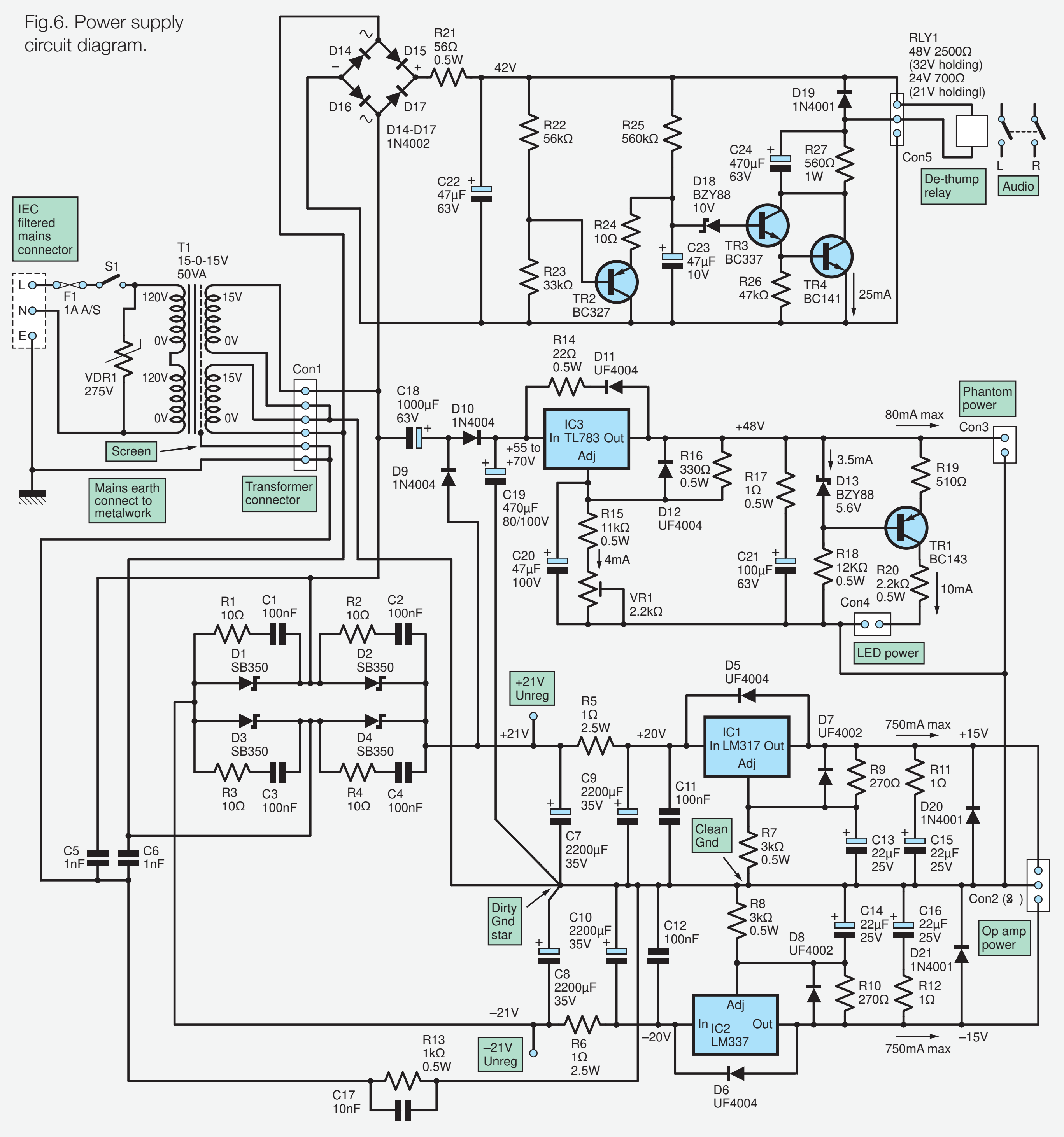


Fig.7. Toroidal transformers run cooler, radiate less magnetic field and mechanical noise than traditional laminated types. The top transformer is unencapsulated, while the lower transformer from RS has the lower noise with a slight size increase.

capacitors makes these problems worse. If cheap ones, having a high ESR (equivalent series resistance) are used, the ringing can be damped out. That said, it is not good design practice to depend on a variable parasitic parameter being 'bad'. A much better approach is to incorporate a specified damping resistor, – typically 0.6Ω to 10Ω – in series with the capacitor.

Circuit description

Fig.5. Shows the basic blocks in the power supply. I'll go through each section describing some of the design aspects. The full circuit diagram is in Fig.6.

Transformer

A 15-0-15V transformer is sufficient for normal op amp rails of ±15V. If a higher voltage is required then the transformer can be increased up to 18-0-18V or 20-0-20V. The best transformers for audio are encapsulated magnetically screened toroids with an interwinding screen between the primary and secondary. These are often made by Avel Lindberg in the UK (sometimes re-badged for RS, see Fig.7). I like to buy them at radio rallies where they sometimes turn up for a few pounds. They seem to last forever, so I also salvage them from scrapped equipment. Only audiophiles seem to appreciate their low-noise qualities, so radio hams charge very little for them. The screen must be connected to mains earth to be effective against radio interference and there is a pin for this on the PCB. If a transformer without a screen is used, placing a couple of capacitors (C5 and C6) between the secondary outputs and mains earth does almost as good a job. They must be kept small (<10nF), or they cause hum spikes.

Rectifier

It is well known that rectifier diodes make a lot of high-frequency noise when they switch off during their conduction cycle. This noise is usually suppressed by wiring snubber capacitors of 10nF to 100nF (C1 to C4) across each diode in a bridge rectifier (D1 to D4). To enhance the snubbing effect and prevent ringing, damping resistors (R1 to R4) are inserted in series with the capacitors. The other rectifiers in the system don't need these capacitors because the currents involved are much lower. The main rectifier uses Schottky diodes for their low voltage drop, which is needed with a 15-0-15V transformer.

Dual-rail op amp supply

This section uses the standard audio LM317/LM337 regulator pair because they are low-noise and cheap. The only problem is their high-current limit of 1.5A; in practice, they usually shut down thermally before this limit is reached. To ensure reasonable output voltage accuracy the potential divider resistors R7 to R10, must have fairly low-resistance values (under 4.7kΩ). This is to accommodate a small but variable current coming out of the regulator's adjust pin which may induce a voltage reflected in the output voltage. In the data sheet these resistors are often half the values I have used to ensure the minimum load current which most regulators require to operate. All the normal protection diodes (D5-D8) have been added for input and output shorts. D20 and D21 are included to prevent start-up problems if there is a momentary short between the plus and minus rails. (In audio circuits there are often decoupling capacitors across the rails which can cause this situation.)

Phantom power voltage tripler

To produce a regulated 48V for phantom power, an unregulated voltage of at least 55V is needed to provide sufficient headroom. Obviously, we can't get this from the 15V transformer output without a bit of circuit trickery. Here I use a standard voltage-doubler circuit consisting of C18, D9, D10 and C19. In fact, since the cathode of D9 is connected to the positive op amp power rail rather than ground, the output of the voltage doubler is actually tripled, giving plenty of headroom. (If there is too much voltage (>70V) on C19 because of higher-voltage transformers, D9 can be switched to 0V). The ground connection of the voltage doubler has to be connected to the 'dirty' star point (see Fig.6) to avoid charging pulses. The ground of the 48V regulator circuit must go to a different ground point, the op amp regulator circuit's dual-rail 0V (see Fig.6), because that is the quiet 'clean' reference ground.

Production of the versatile TL783 high-voltage regulator has resumed recently, courtesy of Texas Instruments, and it is the best device for phantom-power regulators. This is good news for audiophiles since there were no alternatives apart from complicated LM317 circuits or even more complex discrete circuits. The TL783's series-pass transistor is a MOSFET, so it is a thermally robust device.

To accurately set the output voltage, a preset resistor is needed, rather than a fixed potential divider. This is because the output voltage is such a large multiple of the reference voltage that errors are magnified. The current through presets has to be kept to a minimum – high dissipation here is not good for stability and long life. Often, the potential divider resistor values are reduced to double up as a minimum current load for the regulator. On the datasheet this is specified as 15mA. In view of the relatively low output voltage of the regulator (it can be normally used up to 125V) this can be relaxed a bit to 10mA.

LED driver

The need for a minimum load current also means wasted energy if just a bleeder resistor is used. Often, this current is used to run an LED. Since the voltage is high it is possible to run lots of LEDs in series, such as bar graph meters. LEDs used for front-panel indication will occasionally have to be turned off (by using switches to short them out) so some means of stabilising the current through the chain is necessary. This is to avoid changes in the brightness of the remaining LEDs that are illuminated. A constant-current source does the trick, as shown in Fig.8. Note that if some LEDs are too bright relative to others, they will have to have a resistor wired in parallel, bypassing some of the current to dim them down. I have set the current to 10mA, which is sufficient for most modern LEDs. If you are using old, low-efficiency TIL209 LEDs from the 70s, you may have to up it to 20mA by reducing R19 to 240Ω. The constan-current circuit is built around TR1 which needs a clip-on heatsink. R20 relieves it of some of its dissipation burden. A voltage reference of 5.6V is provided by Zener diode D13. This supplies 5V across 510Ω resistor R19, setting the constant current to 10mA. Note that if LEDs are

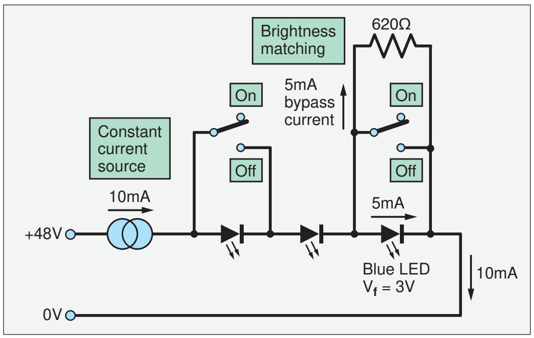


Fig.8. LEDs can be placed in series, fed by a constant-current source. If they need to be extinguished they are shorted with switches.

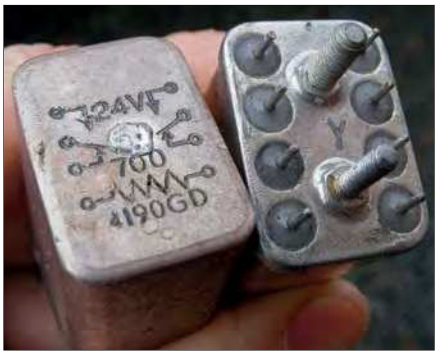


Fig.9. These hermetically sealed 4190 relays from STC have a proven lifetime of around 30 years for audio signal switching. Such relays can be spotted by the glass seals around the pins.

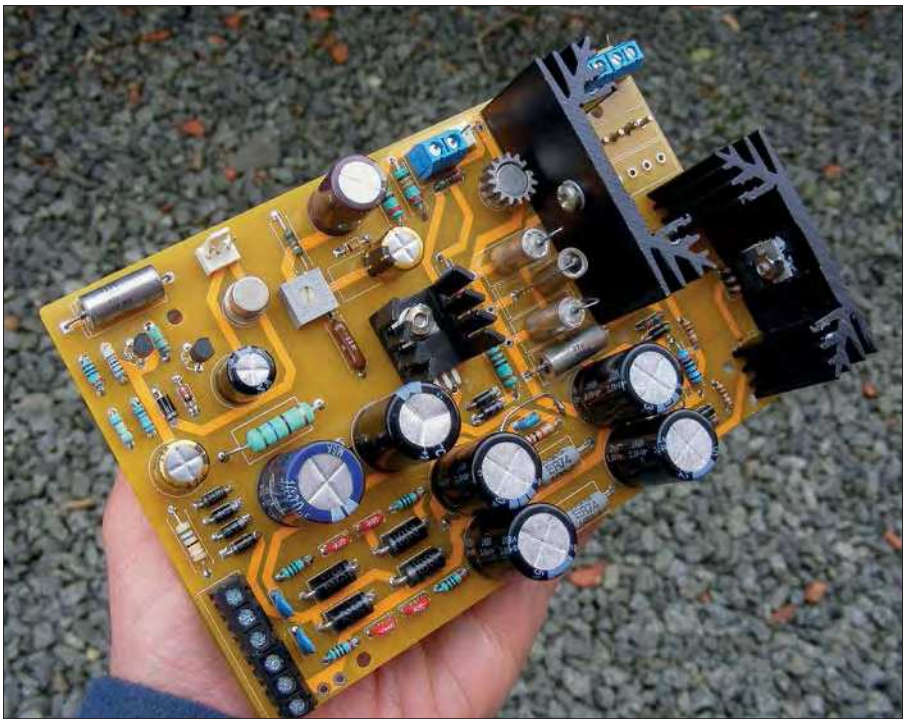


Fig.11. The completed power supply board.

not used or needed then the regulator must still be loaded by shorting out the LED connector.

Muting relay

Another power-related circuit needed is a relay driver. Almost all audio devices require some means of muting switch-on and switch-off noises. Active filters built around op amps, such as those used in the *Analogue Vocoder*, are especially bad, often producing a shriek when turned off. You can imagine the sound of 14 channels squealing at different frequencies simultaneously is pretty horrendous. A double-pole relay or pair of JFETs are often used to mute the audio output lines while these noise sources are active. JFETs have the advantage of being almost a short until biased off, and they consume virtually no power. Relays consume significant power to energise the coil, but they provide the lowest distortion and the best muting. However, unless they are hermetically sealed, the contacts can get oxidised, leading to intermittent signals. Normally, such relays are expensive, but I've got plenty of 'old new stock' inexpensive devices for *PE* readers, as shown in Fig.9 (see my *AO-Shop* ad on p.53 in last month's *PE*).

The relay control circuit has to generate a delay to turn it on and an instant turn-off the moment the power is removed. To get the muting relay to turn off quickly, it's necessary for the circuit to be powered by its own bridge rectifier (D14 to D17) followed by a minimally sized smoothing capacitor (C22), just sufficient to stop the relay buzzing. The switch-on delay of a couple of seconds is determined by R25 and C23. There is also an 11.2V threshold voltage set by a 10V Zener diode (D18) in conjunction with the 1.2V forward drop of the Darlington driver consisting of TR3 and TR4. I could have used a single MOSFET, but bipolar devices work fine. A circuit comprising TR2, which is normally biased off by potential divider R22 and R23 discharges the timing capacitor C23 quickly at turn-off. R24 limits the current protecting the transistor and tantalum capacitor. This ensures the switch-on delay always happens, even if the power is quickly flipped on and off. Finally, a power-saving network, comprising R27 in parallel with large electrolytic C24 ensures sufficient current to pull in the relay on turn-on with just enough current to then hold it. This almost halves the continuous current consumption. This circuit allows the use of 24V 700Ω and 48V 2500Ω relay coil impedance devices.

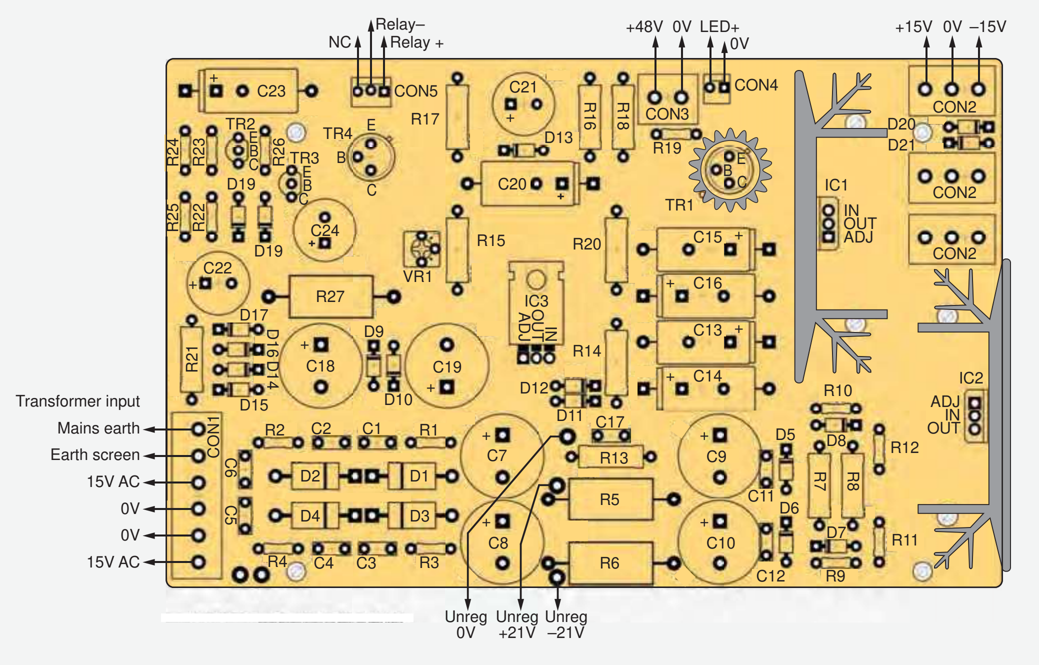


Fig.10. Power supply PCB overlay – note the chunky components, no 64-pin IC packs here! PCB design by Mike Grindle.

Heatsinks

Separate heatsinks are used for each voltage regulator because the insulating washer can then be avoided. These washers cause small TO220 devices to run almost twice as hot because of their additional thermal resistance. When no washers are used, the heatsinks are of course 'live' and

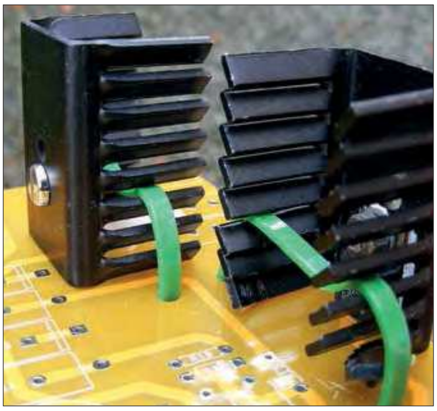


Fig.12. Sometimes, for less arduous applications, smaller pressed heatsinks can be used. If you can keep your finger on the heatsink for a few seconds it will be around 60ºC and within rating.

care has to be taken to avoid shorts. REG1 is at +15V, REG2 at the unregulated input voltage of –21V and REG3 is at +48V.

Large extruded 6.8ºC/W heatsinks are needed for the op amp power regulators (REG1 and REG2) if you plan to draw big currents. The phantom power regulator (REG3) can work with a small clip-on heatsink for normal use. However, if you plan to use up to 20 microphones for a full band, a bigger bolt-down one is needed.

Construction

Like all power supplies, this one uses large parts, so it is much less fiddly than most constructions. The PCB is available from the *PE PCB Service*. The overlay is shown in Fig.10 and the completed board in Fig.11. The main difficulties concern heatsink mounting, which depend to a large extent on what's available. For lower current use, smaller cheaper types can be used, as shown in Fig.12. Remember to use a smear of heatsink compound. Solidity is key with heatsinking, so use spring washers or Nylock nuts to stop the bolts loosening. Also, do secure the heatsinks to the board. To avoid cracked joints, only solder after all the tightening and mechanical adjustment has taken place. Watch out for shorts from the heatsink to the tracks on the upper side of the board. Put Kevlar tape down to provide insulation if necessary.

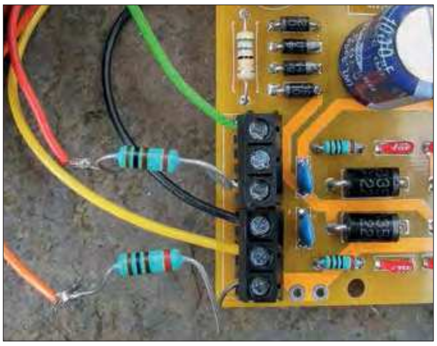


Fig.13. When first turning-on the power supply, wire 100Ω current-limiting resistors between the 15-0-15V secondaries and the PCB. Optionally, these can be replaced later with 2A anti-surge fuses.

Casing

An *external* power supply always gives lowest hum, and when housed in a steel box provides the best screening of the transformers field's hum. If you incorporate the power supply alongside audio equipment in the same housing then always place the transformer as far away as possible from audio inputs. An aluminium case is better here, since magnetic fields are less likely to be radiated into sensitive electronics. Always connect any metalwork to mains earth. I have crammed this power supply into some 1U 19-inch rack cases, but it would be better to use a 2U since there is more headroom above the heatsinks.

Component list

Semiconductors

IC1	LM317T adjustable positive regulator
IC2	LM337T adjustable negative regulator
IC3	TL783 high-voltage regulator, Mouser 595-TL783CKCSE3
D1-D4	SB50 3A 50V (minimum) Schottky diode.
D5-D8, D11, D12	UF4001 ultrafast 1A 50V
D9, D10	1N4004 1A 200V
D15-D17, D19-D21	1N4002 1A 100V
D18	BZY88C5V6 5.6V 400mW Zener 5V6
D19	BZY88C10V 400mW Zener 10V
TR1	BC143 or other medium-power 60V TO5 600mA PNP
TR2	BC327, BC557 or other low-power PNP
TR3	BC337, BC547 low-power NPN
TR4	BC141, 2N2219, BFY51 TO5 40V 600mA NPN

Capacitors

Non-polarised, all 100V ceramic X7R 20% 5mm

C1-C4, C11, C12	100nF
C5, C6	1nF
C17	10nF

Electrolytic capacitors

Note, these components determine the lifespan of the unit. Use the best you can – eg, Kemet, Nichicon

C7-C9	2200µF 35V 16mm diameter radial (Nichicon UPM1V222MHD from Mouser)
C13-C16	22µF 35V axial or radial*
C18	1000µF 63V 16mm radial (Nichicon UPM1J102MHD from Mouser)
C19	470µF 80V or 100V 16mm radial (Nichicon UPM1K471MHD, Mouser)
C20	47µF 50V radial*
C21	100µF 50V radial*
C22	47µF 63V radial*
C23	47µF 16V tantalum axial/ radial
C24	470µF 25V radial*

*Some of the smaller ones could be replaced with solid-aluminium or metal-cased tantalum devices for lower noise and longer life.

Resistors

All 5% tolerance carbon-film, 0.25W unless otherwise stated

R1-R4, R11, R12, R17	1Ω (7 off)
R5, R6	1Ω 2.5W wire-wound/ metal-oxide
R7, R8	3kΩ 0.5W 1% metal-film (Tayda do a 1W version that fits)
R9, R10	270Ω 0.25W 1% metal-film
R13	1kΩ 0.5W
R14	22Ω 0.5W
R15	11kΩ 0.5W
R16	330Ω
R18	12kΩ 0.5W
R19	510Ω
R20	2.2kΩ 0.5W
R21	56Ω 0.5W
R22	56kΩ
R23	33kΩ
R24	10Ω
R25	560kΩ
R26	47kΩ
R27	560Ω 1W metal-oxide
VR1	2.2kΩ horizontal cermet preset, 10mm square Bourns 3386, Helitrim 72P or similar.

Inductive

RLY1	STC 4190 double-pole sealed relay or similar, 24V 700Ω or 48V 2500Ω coil.
T1	Transformer 15-0-15V 2A toroidal Cotswold D1012/A or similar. Avel Lindberg or RS 207-425 screened type if available. (I have second-hand ones in stock – see AOShop advert)

Heatsinks

Redpoint/Avid Thermalloy 436402 6.8ºC/W heatsink RS 402-995 (x2)
TO5 clip-on heatsink
T0220 Redpoint/Avid Thermalloy TV4 or 400043 21ºC/W heatsink RS 402-954

Miscellaneous

PCB from the *PE PCB Service* (AO1-APR22)
M3.5 or 4BA nuts, 10mm bolts and spring washers for mounting the three regulators
Self-tappers No.6 x 6.4mm for large heatsinks RS 504-568.
Power connectors – eg, JAE SRC-N6A13-5S-A534 from Mouser, suitable for Soundcraft mixer
0.2-inch screw connectors
'Molex' 0.1 inch 3-way and 2-way crimp headers and plugs

Testing

Since power supplies have many electrolytic capacitors and diodes, there is always the risk of an 'explosion' if connection are reversed. It's a good idea to place 100Ω 0.5W current-limiting resistors in series with the 15V transformer input leads, as shown in Fig.13. There should be no load on the power supply at this stage. If they start to smoke and there's no voltage on any of the unregulated output pins, something is wrong! The good thing is you won't have a burnt-out £20 transformer or lost an eye – yes, *always* wear eye protection for 'first power up' of any circuit!

Full load check

It is sensible to test a power supply by drawing the full current expected plus a little extra from all outputs and looking at the output with a 'scope set to 5mV/div AC. This should also be done to check for heating effects and possible loss of regulation which can happen if the input voltage to the regulators falls too low, allowing ripple to break through. Noise should just be a smooth hiss at around $5mV_{pk\text{-}pk}$ with no hum spikes. Suggested brutal test load resistors are 560Ω 10W for the 48V and 20Ω 10W for the ±15V rails.

Outdoor musical events are often powered by generators, prone to 'mains' voltage dips. To check fully for this, I feed in the minimum expected mains voltage (say 215V) using a variac or buck transformer. In this case, it will be necessary to use an 18-0-18V transformer and reduce the maximum load current by 20%.

That's all folks

We haven't got space or time this month to integrate this power supply into the *Analogue Vocoder* so that will have to be covered next time.

ESR Electronic Components Ltd

All of our stock is RoHS compliant and CE approved. Visit our well stocked shop for all of your requirements or order on-line.

We can help and advise with your enquiry, from design to construction.

3D Printing • Cable • CCTV • Connectors • Components • Enclosures • Fans • Fuses • Hardware • Lamps • LED's • Leads • Loudspeakers • Panel Meters • PCB Production • Power Supplies • Relays • Resistors • Semiconductors • Soldering Irons • Switches • Test Equipment • Transformers and so much more...

Monday to Friday 08:30 - 17.00, Saturday 08:30 - 15:30

Station Road
Cullercoats
North Shields
Tyne & Wear
NE30 4PQ

Tel: 0191 2514363 sales@esr.co.uk www.esr.co.uk

Fluke/Philips PM3092 Oscilloscope
2+2 Channel 200MHz Delay TB,
Autoset etc – £250

STEWART OF READING
17A King Street, Mortimer, near Reading, RG7 3RS
Telephone: **0118 933 1111** Fax: **0118 933 2375**
USED ELECTRONIC TEST EQUIPMENT
Check website **www.stewart-of-reading.co.uk**
(ALL PRICES PLUS CARRIAGE & VAT)
Please check availability before ordering or calling in

HP 54600B Oscilloscope
Analogue/Digital Dual Trace 100MHz
Only £75, with accessories £125

LAMBDA GENESYS	PSU GEN100-15 100V 15A Boxed As New	£400	Marconi 2305	Modulation Meter	£250
LAMBDA GENESYS	PSU GEN50-30 50V 30A	£400	Marconi 2440	Counter 20GHz	£295
IFR 2025	Signal Generator 9kHz – 2.51GHz Opt 04/11	£900	Marconi 2945/A/B	Communications Test Set Various Options	POA
IFR 2948B	Communication Service Monitor Opts 03/25 Avionics	POA	Marconi 2955	Radio Communications Test Set	£595
IFR 6843	Microwave Systems Analyser 10MHz – 20GHz	POA	Marconi 2955A	Radio Communications Test Set	£725
R&S APN62	Syn Function Generator 1Hz – 260kHz	£295	Marconi 2955B	Radio Communications Test Set	£800
Agilent 8712ET	RF Network Analyser 300kHz – 1300MHz	POA	Marconi 6200	Microwave Test Set	£1,500
HP8903A/B	Audio Analyser	£750 – £950	Marconi 6200A	Microwave Test Set 10MHz – 20GHz	£1,950
HP8757D	Scaler Network Analyser	POA	Marconi 6200B	Microwave Test Set	£2,300
HP3325A	Synthesised Function Generator	£195	Marconi 6960B	Power Meter with 6910 sensor	£295
HP3561A	Dynamic Signal Analyser	£650	Tektronix TDS3052B	Oscilloscope 500MHz 2.5GS/s	£1,250
HP6032A	PSU 0-60V 0-50A 1000W	£750	Tektronix TDS3032	Oscilloscope 300MHz 2.5GS/s	£995
HP6622A	PSU 0-20V 4A Twice or 0-50V 2A Twice	£350	Tektronix TDS3012	Oscilloscope 2 Channel 100MHz 1.25GS/s	£450
HP6624A	PSU 4 Outputs	£400	Tektronix 2430A	Oscilloscope Dual Trace 150MHz 100MS/s	£350
HP6632B	PSU 0-20V 0-5A	£195	Tektronix 2465B	Oscilloscope 4 Channel 400MHz	£600
HP6644A	PSU 0-60V 3.5A	£400	Farnell AP60/50	PSU 0-60V 0-50A 1kW Switch Mode	£300
HP6654A	PSU 0-60V 0-9A	£500	Farnell XA35/2T	PSU 0-35V 0-2A Twice Digital	£75
HP8341A	Synthesised Sweep Generator 10MHz – 20GHz	£2,000	Farnell AP100-90	Power Supply 100V 90A	£900
HP83630A	Synthesised Sweeper 10MHz – 26.5 GHz	POA	Farnell LF1	Sine/Sq Oscillator 10Hz – 1MHz	£45
HP83624A	Synthesised Sweeper 2 – 20GHz	POA	Racal 1991	Counter/Timer 160MHz 9 Digit	£150
HP8484A	Power Sensor 0.01-18GHz 3nW-10µW	£75	Racal 2101	Counter 20GHz LED	£295
HP8560E	Spectrum Analyser Synthesised 30Hz – 2.9GHz	£1,750	Racal 9300	True RMS Millivoltmeter 5Hz – 20MHz etc	£45
HP8563A	Spectrum Analyser Synthesised 9kHz – 22GHz	£2,250	Racal 9300B	As 9300	£75
HP8566B	Spectrum Analsyer 100Hz – 22GHz	£1,200	Solartron 7150/PLUS	6½ Digit DMM True RMS IEEE	£65/£75
HP8662A	RF Generator 10kHz – 1280MHz	£750	Solatron 1253	Gain Phase Analyser 1mHz – 20kHz	£600
Marconi 2022E	Synthesised AM/FM Signal Generator 10kHz – 1.01GHz	£325	Solartron SI 1255	HF Frequency Response Analyser	POA
Marconi 2024	Synthesised Signal Generator 9kHz – 2.4GHz	£800	Tasakago TM035-2	PSU 0-35V 0-2A 2 Meters	£30
Marconi 2030	Synthesised Signal Generator 10kHz – 1.35GHz	£750	Thurlby PL320QMD	PSU 0-30V 0-2A Twice	£160 – £200
Marconi 2023A	Signal Generator 9kHz – 1.2GHz	£700	Thurlby TG210	Function Generator 0.002-2MHz TTL etc Kenwood Badged	£65

HP/Agilent HP 34401A Digital
Multimeter 6½ Digit £325 – £375

HP33120A	Function Generator 100 microHz – 15MHz	£350
HP53131A	Universal Counter 3GHz Boxed unused	£600
HP53131A	Universal Counter 225MHz	£350
Audio Precision	SYS2712 Audio Analyser – in original box	POA
Datron 4708	Autocal Multifunction Standard	POA
Druck DPI 515	Pressure Calibrator/Controller	£400
Datron 1081	Autocal Standards Multimeter	POA
ENI 325LA	RF Power Amplifier 250kHz – 150MHz 25W 50dB	POA
Keithley 228	Voltage/Current Source	POA
Time 9818	DC Current & Voltage Calibrator	POA

Marconi 2955B Radio
Communications Test Set – £800

Max's Cool Beans

By Max the Magnificent

Flashing LEDs and drooling engineers – Part 26

A few days ago, as I pen these words, I was watching a classic movie on TV with my wife (Gina the Gorgeous) and my son (Joseph the Commonsense Challenged). The film's title, which is on the tip of my whatsit, was something like *Departed with Flatulence* (I'll remember the real name and post it at the end of this article). There comes a point about two hours in when our starving heroine finds a turnip in a field, scarfs it down in a very unladylike manner, has an upset tummy, and then holds the remains of the turnip up to the heavens while proclaiming, 'I swear to God that I'll never be hungry again!'

At this point there's a crescendo of music and an 'Intermission' sign appears on the screen, just like it did when this movie was first presented in cinemas. 'Well, that was the worst ending to a film ever', said Joseph, who – it turned out – was blissfully unaware as to the meaning of the word 'Intermission'. You should have seen his face when I reassured him that we had only just reached the halfway point and we still had the same amount of film to go. I found myself talking to a hole in the air where a boy had been. I was most impressed. I had no idea he could bounce out of a room so fast. And, speaking of bouncing…

You turn me on (and off)

One of the topics on which we touched in my previous column (*PE*, March 2022) was switch bounce. That is, when you close (make) or open (break) a switch like a pushbutton switch or a toggle switch, rather than providing a clean transition between logic 0 and logic 1 values, it can bounce back and forth anywhere up to 100+ times before settling into its new state.

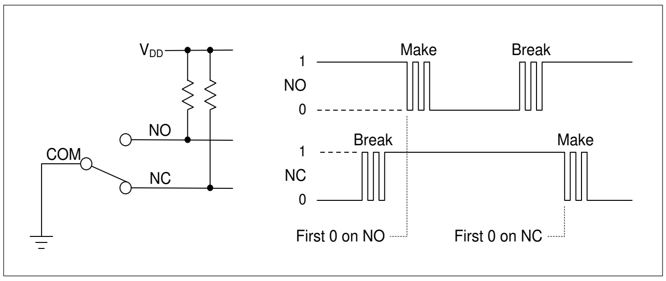


Fig.1. Single-pole, double-throw (SPDT) switch.

For the purposes of those discussions, we focused on a single-pole, single-throw (SPST) switch and we discussed two software solutions that can be used to debounce (clean up) the signal coming from the switch. The first of these solutions involved the use of a cunning counter; the second employed a sumptuous shift register.

Personally, I find the topic of switches in general, and debouncing them in particular, to be extremely interesting and thought-provoking. There's always something new to learn. For example, what exactly is the definition of a switch? Well, in its simplest terms, we might describe a switch as being an electrical component that can 'make' or 'break' an electrical circuit (which explains the 'make' and 'break' nomenclature we've been using), thereby interrupting the electrical current or diverting it from one conductor to another.

Something else in an electrical engineer's argot that can confuse beginners are the terms 'poles' and 'throws'. In the context of switches, the number of poles defines the number of distinct circuits a switch can control. Thus, a switch with only one pole ('single pole') can control only one circuit; a switch with two poles ('double pole') can control two separate circuits; and so on. Meanwhile, the term 'throws' defines the number of contacts to which each of the switch's poles can be connected. Thus, if a switch has only one throw ('single throw'), then each of its poles can be connected to only one contact; if a switch has two throws ('double throw'), then each of its poles can be connected to two contacts; and so forth.

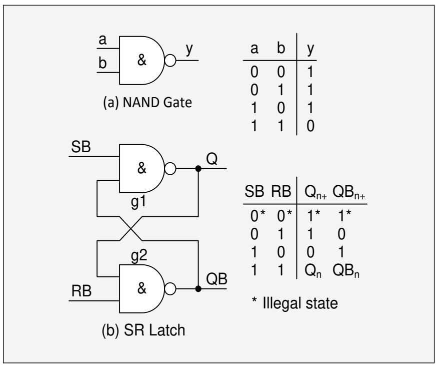
When it comes to things like pushbutton switches and toggle switches, we tend to think of the throws as being the fixed contacts and the poles as being the moving contacts; that is, the poles are the contacts that are connected to the moving parts of the switch. In turn, this leads us to consider the difference between the terms 'contacts' and 'terminals'. The way I think about this is that the contacts are internal to the switch, while the terminals are the equivalents to which you connect your wires outside the switch, and that each contact will be connected to one or more terminals.

Double the fun

Let's up the ante a little by turning our attention to consider a single-pole, double-throw (SPDT) switch (Fig.1). In this case, the pole is usually referred to as being the 'common' terminal, and may therefore be annotated as COM. One of the throws will be described as being 'normally open' (NO), while the other will be characterised as being 'normally closed' (NC). These terms are based on the state of the terminals when the switch is in its inactive/off state (whatever that means).

Just to increase the fun and frivolity (if that's even possible), the SPDT switches most of us tend to use in our everyday lives, and the ones on which we are focusing here, are classed as 'break-before-make' (BBM). This means that when the switch is deactivated or activated, the pole contact is disconnected from its current throw contact before being connected to its new throw contact. There are also 'make-before-break' (MBB) counterparts in which the new connection is made before the old connection is broken, but we tend to save those switches for special occasions.

The interesting thing to note about a BBM SPDT is that – due to its physical construction – the contact being broken will always have finished bouncing some time before the contact that is being made commences to bounce. This nugget of knowledge makes it extremely easy to implement a simple and robust software debounce solution. Let's assume that the switch commences in its inactive state with NO = 1 and NC = 0, as illustrated in Fig.1. In this case, all we have to do is look for the first time we see NO = 0 and NC = 1, which tells us that the NC contact has finished bouncing and the NO contact is just starting to transition (in reality, we can just look for the first time NO = 0). Similarly, once the switch



a	b	y
0	0	1
0	1	1
1	0	1
1	1	0

SB	RB	Q_{n+}	QB_{n+}
0*	0*	1*	1*
0	1	1	0
1	0	0	1
1	1	Q_n	QB_n

* Illegal state

Fig.2. NAND gate and SR latch.

is in its active state with NO = 0 and NC = 1, all we have to do is look for the first time we see NO = 1 and NC = 0, which tells us that the NO contact has finished bouncing and the NC contact is just starting to transition (in reality, we can just look for the first time NC = 0).

For reasons that will soon become apparent, the important thing to observe here is that there are only ever three combinations of 0s and 1s on NO and NC: 10, 11, and 01. That is, we never see the 00 case. Also, the switch may bounce back and forth between the 10 and 11 states, and it may bounce between the 11 and 01 states, but it will never bounce between the 10 and 01 states. Another way to look at this is that the 11 state acts as a sort of buffer zone between the 10 and 01 states. Thus, if we ignore any bounces, the main state transitions will be: 10 -> 11 -> 01 -> 11 -> 10 -> ... and so on.

Handy hardware

It's worth reminding ourselves that we didn't use to care about switch bounce – most people didn't even know it existed – prior to our creating high-speed electronic systems. Even today, when we turn a light on in the front room, for example, any bouncing takes place far too quickly for us to perceive it. It was only when we started to build devices like computers that switch bounce reared its ugly head.

In the golden age of mainframe computers circa the 1960s, the combination of SPDT switches and the software debounce solution discussed above formed one of the *crème de la crème* solutions used to address switch bounce. The only problem with this approach was that each switch required two of the computer's input pins. This was a pain when you remember that switches were one of the main ways to communicate with a mainframe computer, so there could be hundreds of the little scamps. One way to tackle this was to employ some handy hardware in the form of two 2-input NAND gates (Fig.2a) used to form an SR ('Set-Reset') latch (Fig.2b).

With regard to the 'n' and 'n+' annotations in our truth table, we are using 'n+' to represent our new state (ie, the state at time 'now + 1'), while 'n' represents the current state (ie, the state at time 'now'). Don't worry, this will make more sense in a moment.

An SR latch employs feedback (that is, feeding its outputs back as inputs) to remember its previous state. This type of circuit is one of the simplest forms of memory elements we can make. Unfortunately, even though our implementation uses only two 2-input NAND gates, wrapping our brains around how it works can be tortuously tricky for those who are new to this sort of thing. On the other hand, once we *have* wrapped our brains around the workings of the SR latch, it makes understanding other circuits of this ilk much, much easier, so let's take a deep breath and ponder things step-by-step.

Let's start with the fact that a 0 on any of a NAND gate's inputs causes a 1 on its output. It's only when all of the NAND gate's inputs are 1 that its output will be 0. The next point to note is that, in the case of our implementation of the SR latch, the SB ('set') and RB ('reset') inputs are active-low; that is, they are active when they are in a 0 state. This explains why we've used the 'B' suffix meaning 'bar' because an alternative way to write these signal names would be as $\overline{S}$ and $\overline{R}$ characters, each with a horizontal line (also referred to as a 'bar') over the top.

Now let's assume that we've just powered our circuit up and that both SB and RB are in their inactive 1 states, which I've indicated in red (Fig.3a). In this case, for reasons that will become clear momentarily, we currently have no idea as to the values on the Q and QB outputs, which I've represented in orange and as 'X' in the figure.

In Fig.3b we place the SB input in its active 0 state. This actually triggers a series of actions as follows. First, since SB is 0, the output of the upper NAND gate, which is connected to the Q output, will be driven to a logic 1. Next, since the Q output feeds back to the lower NAND gate, this means that both of the inputs to the lower NAND gate are now 1. In turn, this means that the output from the lower NAND gate, which is connected to the QB output, will be driven to 0.

The really important thing to note here is that the QB output feeds back to the upper NAND gate, which means that both of this gate's inputs are now 0. The reason this is so important becomes apparent in Fig.3c when we return the SB input to its inactive 1 state. The fact that QB is 0 ensures that Q is 1, and the fact that both Q and RB are 1 ensures that QB stays at 0, at which point we could return to the start of this sentence.

In Fig.3d we place the RB input in its active 0 state. Once again, this triggers a series of actions as follows. First, since RB is 0, the output of the lower NAND gate, which is connected to the QB output, will be driven to a logic 1. Next, since the QB output feeds back to the upper NAND gate, this means that both of the inputs to the upper NAND gate are now 1. In turn, this means that the output from the upper NAND gate, which is connected to the Q output, will be driven to 0.

In this case, the fact that the Q output feeds back to the lower NAND gate means that both of this gate's inputs are now 0. Thus, when we return input RB to its 1 state (Fig.3e), the fact that Q is 0 ensures that QB is 1, and the fact that both QB and SB are 1 ensures that Q stays at 0.

All of this explains why we show the state in which both SB and RB are 0 as being illegal in the truth table for the RS latch (Fig.2b). We can certainly place both of these inputs in their active 0 states, which will cause both the Q and QB outputs to be driven to 1 (remember any 0 on the input to a NAND gate will result in a 1 on its output). Furthermore, there won't be any issues if we return SB to its inactive state and then wait awhile before we return RB to its inactive state, or vice versa. The problem arises if we return both SB and RB to their inactive states simultaneously. In this case, much like tossing a spinning coin into the air, the Q and QB outputs can end up toggling back and forth until the latch eventually, and randomly, settles one way or the other. Alternatively, the Q and QB outputs may end up at an intermediate voltage level, which could cause all sorts of nasty things to happen.

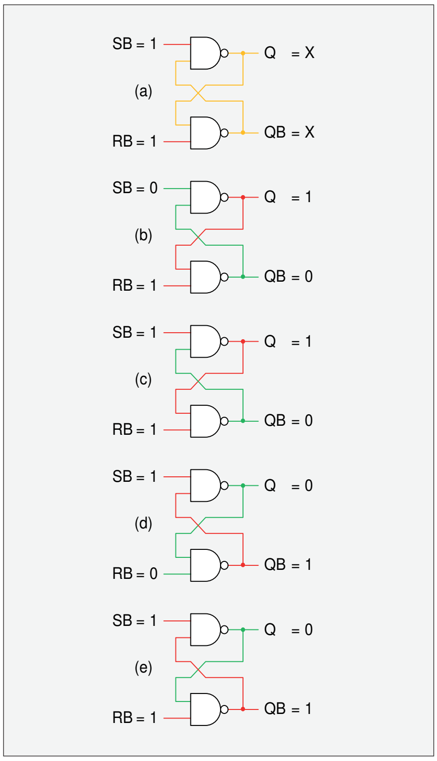


Fig.3. Different states of an SR latch.

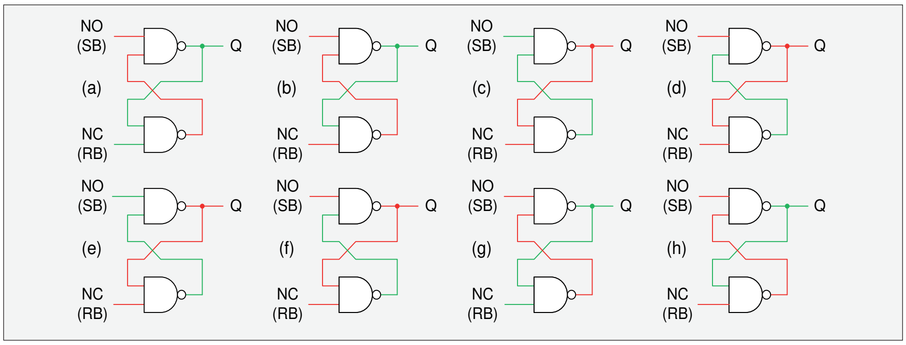


Fig.4. Using an SR latch to debounce an SPDT switch.

Tra-la!

This is where we bring everything together. Let's suppose that we connect the NO and NC outputs from our SPDT toggle switch to the SB and RB inputs of our SR latch, respectively (Fig.4). As part of this, we'll remove the QB output from the latch and keep only the Q output, which is what we will eventually connect to the input of our computer. Earlier, we noted that, as seen in Fig.1, the switch's NO and NC outputs are never both 0 at the same time. In turn, this means we never encounter the RS latch's illegal state as seen in Fig.2b (it's almost as though we had a plan).

Once again, let's assume that the switch commences in its inactive state with NO (SB) = 1 and NC (RB) = 0, as illustrated in Fig.1 and Fig.4a. Based on our earlier discussions, we know that the 0 on NC causes a 0 on Q. When we start to activate our switch (turn it on), the first thing that will happen is that the NC signal will change from 0 to 1 (Fig.4b), but this will have no effect on the output Q. Furthermore, it doesn't matter if the NC signal bounces back and forth between 0 (Fig.4a) and 1 (Fig.4b) because Q will remain at 0. Eventually, the NC signal will finish bouncing, leaving the switch as shown in Fig.4b, after which the NO signal will start bouncing. It's only when the NO signal first transitions to 0 that the Q terminal will transition to 1 (Fig.4c), which will cause the output from the lower NAND gate to transition to 0. Even if the NO signal bounces back to 1 (Fig.4d), the Q output will remain at 1. Eventually, the NO signal will finish bouncing, leaving the state of the switch as shown in Fig.4c.

The switch is now in its active state with NO = 0, NC = 1, and Q = 1 (Fig.4e). When we start to deactivate our switch (turn it off), the first thing that will happen is that the NO signal will change from 0 to 1 (Fig.4f), but this will have no effect on the output Q. In this case, it doesn't matter if the NO signal bounces back and forth between 0 (Fig.4e) and 1 (Fig.4f) because Q will remain at 1. Eventually, the NO signal will finish bouncing. It's only when the NC signal first transitions to 0 that the output from the lower NAND gate will be driven to 1, which will result in the Q terminal being driven to 0 (Fig.4g). Even if the NC signal bounces back to 1 (Fig.4h), the Q output will remain at 0. Eventually, the NC signal will finish bouncing, leaving the state of the switch as shown in Fig.4g.

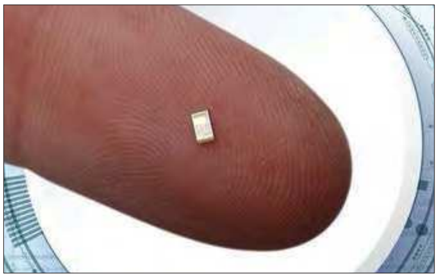


Fig.5. The extra-tiny Ultrasonic TouchPoint Z sensor from Ultrasense.

It's only when you stop to think about it that you realise that this hardware implementation debounces the signal from the switch in *exactly the same way* as the software solution we discussed earlier. Tra-la!

A little bit of history

As we noted earlier, 'mainframe computers circa the 1960s might boast hundreds of switches. We also made mention of the fact that SPDT switches – especially when combined with SR latches – formed the '*crème de la crème* switch debounce solution'. Why didn't the designers of these 'big iron' machines omit the SR latches, use cheaper SPST switches, and debounce the signals in software?

One reason was that these computers were already ferociously expensive – in the US, for example, it could cost thousands of dollars a month just to rent one of these machines, or around a million dollars to buy one – so price was not an issue with respect to splashing out the cash for a few hundred switches. On the other hand, even though 1960s mainframes were physically large, they were computationally challenged by today's standards, with limited memory and clock cycles. Consider the IBM System/360 Model 30, for example. Introduced in 1964, this behemoth was equipped with only 8 to 64KB of memory and could perform only 34,500 instructions per second. Not surprisingly, no one wanted to waste their precious memory or clock cycles debouncing switches.

When the use of microprocessor-based personal computers started to explode circa the late-1970s and early 1980s, there were a lot of people with hardware design experience, but relatively few experienced software developers. Furthermore, what software developers there were rarely understood the problem of switch debounce, so it was left to the hardware folks to sort things out. Since money was a consideration with these machines, designers often employed cheap-and-cheerful SPST switches coupled with resistor-capacitor (RC) delay circuits, which we discussed in an earlier *Cool Beans* column (*PE*, April 2019).

As microprocessor-based computers grew in capacity and performance, coupled with the exponential growth in microcontroller-based embedded systems, more and more people learned to program, and it became increasingly common to debounce switches in software. Even today, however, some people prefer to use hardware debounce techniques.

Welcome to the 21st Century

On the one hand... I love traditional pushbutton and toggle switches. For example, I'm currently watching the neo-noir science fiction TV series *Cowboy Bebop* on Netflix – not the original Japanese anime version, but rather the 2021 live action remake (**https://imdb.to/35psfDC**). Although this is set in the year 2171, the technology has a strong steampunk aesthetic that really tickles my fancy, as it were. For example, our heroes – Jet Black and Spike Spiegel – spend a lot of time flying around in their spaceship, the *Bebop*, whose control panels are unrestrainedly festooned with a cornucopia of vintage toggle switches.

On the other hand... when you come to think about it, if you look around at all of the amazing electronic systems that surround us, don't you sometimes think to yourself, 'Pushbutton and toggle switches are so 20th Century!' I mean to say, pushbutton and toggle switches of the type with which we are familiar originally appeared on the scene circa 1880 and 1916, respectively. In turn, this means that we are controlling our uber-cool cunning contraptions using devices developed 140+ years ago (in the case of pushbutton switches) and 125+ years ago (in the case of toggle switches). Even worse, we are using devices whose capricious characteristics – including the fact that they bounce like deranged ping pong balls – demand that we expend an inordinate amount of time and effort making them behave the way we want them to in the first place.

Isn't it perhaps time that we started to indulge ourselves with a technology that's a tad more Jetson-esque, as it were? 'But what is the alternative?' I hear you cry. Well, I'm glad you asked, because I was

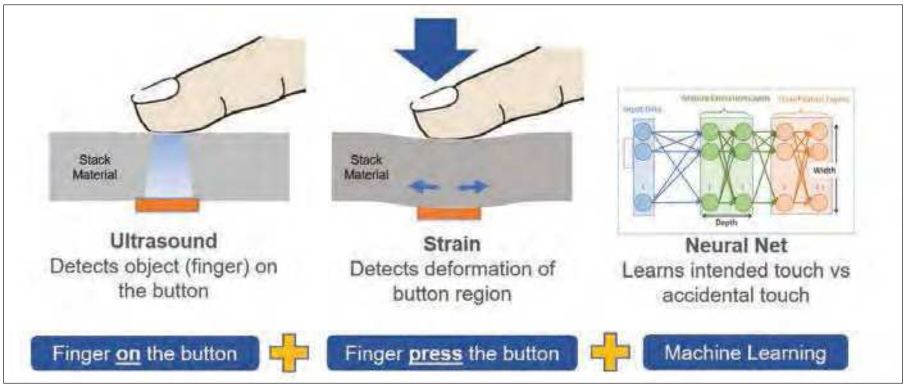


Fig.6. Truly a switch for the 21st Century

recently chatting to the guys and gals at a company called Ultrasense Systems (see: **https://bit.ly/3APS37B**).

These little rascals have developed a teeny-tiny ultrasonic sensor called the TouchPoint Z, which makes a grain of rice look large by comparison (Fig.5).

This sensor can be mounted behind, or embedded in, a control surface of any practical thickness, because its ultrasonic signal can penetrate hard plastic up to 5mm thick and stainless steel up to 2mm thick. Now, there are other ultrasonic sensors around, so what makes this one different. Well, in addition to the ultrasonic transceiver, it also features four tiny piezo strain gauges that allow it to detect deformations as small as 100nm in the surrounding material. But wait, there's more, because it also includes a microcontroller that allows it to communicate with a host system using 2-wire I^2C or UART interfaces. And, just to add a great big dollop of whipped cream on top of the metaphorical cake, the latest generation also boasts a neural touch engine (NTE). By supporting native machine learning (ML), the NTE can be taught to differentiate between intended vs. accidental touches (Fig.6). It probably goes without saying that the one thing this little beauty doesn't give us is any form of switch bounce.

I was reading an article on *Consumer Reports* about Tesla's weird and wonderful rectangular steering yoke with which they've replaced the traditional round steering wheel (**https://bit.ly/32R8fsF**). This yoke features a collection of capacitive sensors. I think the current state of play on the report card is 'Could do better!' For example, one driver noted 'I accidentally washed the windshield and honked the horn at innocent roadgoers while making turns.' Suffice to say, this wouldn't be a problem if they were using ultrasonic sensors from Ultrasense (just wait until Elon reads this column).

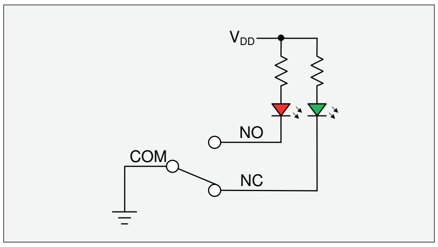


Fig.7. Why, the LEDs are over here!

But where are the flashing LEDs?

I fear that, having seen its title *Flashing LEDs and drooling engineers*, newcomers to this column may be saying to themselves, 'But where are the flashing LEDs?'.

Well, wandering off into the weeds as is my wont, I might note that – in addition to innovative spelling, punctuation and grammatical constructions – technical manuals originating from different countries can have very different styles. Manuals originating in Israel, for example, often have a substantial Shakespearean sense of wordplay, along the lines of, 'But where are the LEDs?' you ask,' closely followed by, 'Why, the LEDs are over here!'

If we were to take our original SPDT switch-based circuit (Fig.1) and add two LEDs (Fig.7) such that the red and green LEDs are connected to the switch's NO and NC terminals, respectively, then the green LED will be illuminated when the switch is in its inactive/off position, while the red LED will light up when the switch is in its active/on position. If you imagine toggling the switch back and forth as fast as you can, then you can also imagine me saying, 'Why, the flashing LEDs are over here!' You're welcome.

All is revealed

I remember now. The title of the movie mentioned at the start of this column was – drum roll – *Gone with the Wind*.

Cool bean Max Maxfield (Hawaiian shirt, on the right) is emperor of all he surveys at **CliveMaxfield.com** – the go-to site for the latest and greatest in technological geekdom.

Comments or questions? Email Max at: **max@CliveMaxfield.com**

PoLabs
For more information or software download please visit
www.poscope.com/epe
PoKeys
Connect, Control
- USB
- Ethernet
- Web server
- Modbus
- CNC (Mach3/4)
- IO
- PWM
- Encoders
- LCD
- Analog inputs
- Compact PLC
Stepper motor drivers
Drive
- up to 256 microsteps
- 50 V / 6 A
- USB configuration
- Isolated
- up to 32 microsteps
- 30 V / 2.5 A
PoScope Mega1+
PoScope Mega50
Measure
- up to 50MS/s
- resolution up to 12bit
- Lowest power consumption
- Smallest and lightest
- 7 in 1: Oscilloscope, FFT, X/Y, Recorder, Logic Analyzer, Protocol decoder, Signal generator

Make it with Micromite

Phil Boyce – hands on with the mighty PIC-powered, BASIC microcontroller

Part 35: Mini-Project: iButton-controlled Electronic Door Lock

This month, we are going to begin the *iButton Electronic Door Lock* (*iEDL*) mini-project. The idea is to bring together the topics covered in the last three articles (January – March 2022) of the *Make it with Micromite* (*MIWM*) series and put all we learnt to practical use.

Do note that we will not be describing details of the various electromechanical door solenoids (or 'holding magnets') that are typically used for locking/releasing a door. Our *iEDL* will simply provide a set of change-over relay contacts that can either be interfaced with an existing electromechanical lock, or alternatively used to control any new mechanical-hardware that needs to be installed.

The concept behind the *iEDL* is simple – issue an iButton key to anyone that is permitted to open (ie, access) one or any selected door(s) in a building. For example, doors around the home, in an office environment or even a large commercial business premises. An *iEDL* is located next to each restricted door, and on presenting a valid iButton key, a person can unlock the door. This might sound the same as any other simple electronic door lock, so what makes the *iEDL* so special?

iEDL features

The simple answer is there's a Micromite at the heart of our *iEDL*, so it can easily be programmed in MMBASIC with as many features and options that we can dream up; and thus we can create a much more intelligent and flexible lock. For example:

- Incorporate a TFT display in the *iEDL*, so the name of the person trying to gain access can be shown – this creates a better user experience
- If the *iEDL* display is also a touchscreen, then for extra security a PIN can also be requested (in addition to presenting the iButton-key)
- Using a 'master Configuration Button key', the *iEDL* can be triggered to enter a configuration mode. The ability to switch the *iEDL* to behave as a configuration tool avoids the need to link the *iEDL* to a computer, and creates a standalone solution
- In configuration mode, the name of the keyholder can be programmed into an EEPROM iButton key
- In configuration mode, an iButton key can be reassigned to a new user or wiped from the system
- In configuration mode, an *iEDL* can be programmed with a descriptor (such as 'Staff Entrance'), and this can be shown on the *iEDL* display
- This project really comes to life when more than one *iEDL* is implemented on a site. Each *iEDL* can be programmed with a single Category ID, which can be set when the *iEDL* is in configuration mode. If access settings (ie, one or more Category IDs) are also programmed into an EEPROM key (as well as the keyholder's name), then a user can be granted access to certain category doors, but denied access to other category doors. This also means that an *iEDL* does not need to contain a list of valid key numbers – instead, the single Category ID programmed into the *iEDL* is checked against the Category IDs programmed into the presented EEPROM iButton, and access is then either permitted, or denied, accordingly.

Micromite code

The code in this article is available for download from the *PE* website.

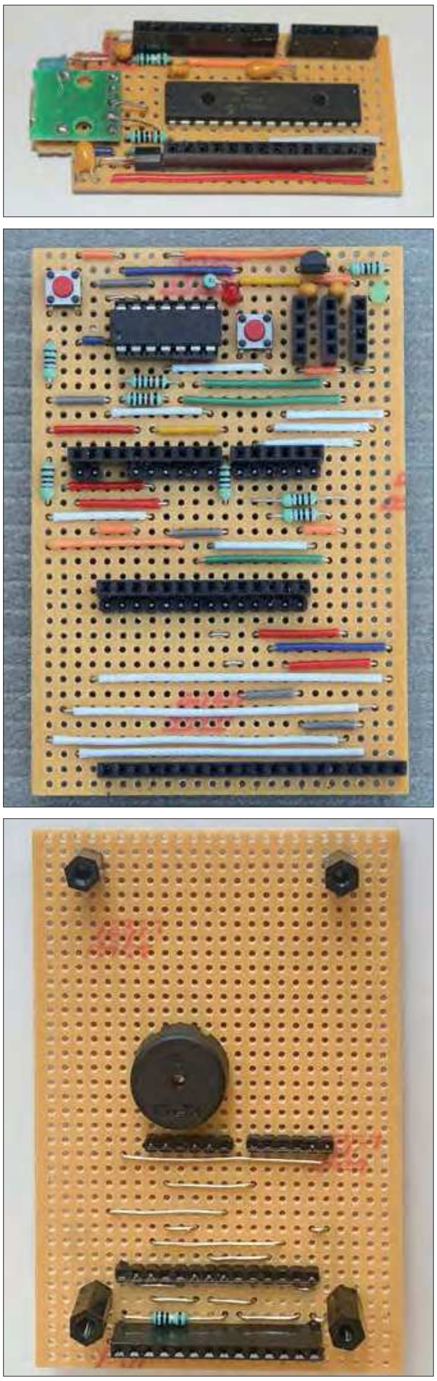


Fig.1. Three *MIWM* modules are used in the *iEDL* prototype: (from top to bottom) a) The *Micromite Keyring Computer (MKC);* b) The *Development Module (DM);* c) The *TFT-adaptor* (complete with piezo sounder).

From the above list of examples, it can be seen how the functionality of the solution could be made to behave in an intelligent way. All that is required is a change to the main program-code in the *iEDL*; with the hardware remaining exactly the same. We will be incorporating many of the above features (next month), but first we need to discuss how to assemble an *iEDL*.

Hardware

As mentioned above, the *iEDL* hardware is built around a Micromite; so, in keeping with the original concept of this series, the prototype *iEDL* was based on the following modules:

- *Micromite Keyring Computer* (*MKC* – from *MIWM Part 2*, *PE* March 2019) – see Fig.1a
- *Development Module* (*DM* – from *MIWM Part 3*, *PE* April 2019) – see Fig.1b
- *TFT-adaptor* (*MIWM Part 15*, *PE* April 2020) – see Fig.1c. This contains the touchscreen, as well as a piezo sounder
- iButton reader (from *MIWM Part 32*, *PE* Jan 2022) – see Fig.2
- LED (if not built into the iButton reader), plus current-limiting resistor
- Low-cost relay module – see Fig.3.

Note that the *MKC* and *DM* can be substituted with an alternative Micromite module, as has been discussed elsewhere in this series. For example, it is possible to use an Explore28 (or Explore64), but this would need to be hard-wired to a touch-TFT. To avoid having to hard-wire a TFT, it is better to use a 28-pin *Micromite BackPack* module (version 2 – see *PE*, May 2018; PCBs are available from the *PE PCB Service*) which would just need the required iButton reader, LED, piezo sounder and relay module to be connected.

Once you have chosen your preferred Micromite to build the *iEDL*, you just need to add the required external parts. To assist with assembly, please refer to Fig.4; this summarises the Micromite pin number connections to all the other parts of the *iEDL*.

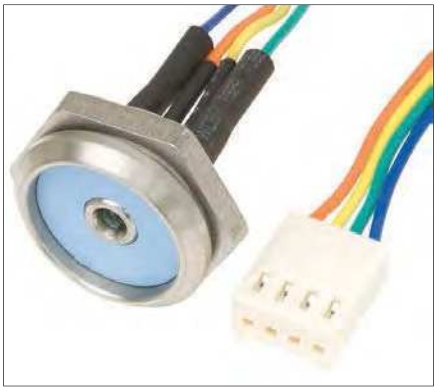


Fig.2. The DS9092L is our preferred iButton reader. It contains a centre-mounted LED which assists in locating the exact position for where to present the iButton.

A point worth noting here is that the cost of an *iEDL* is around £50 (uncased), so it is a much cheaper solution compared to commercially available door-access systems – but, can be fully configured (ie, programmed) to behave as you need it to. (*Do note that questions relating to compatibility of the iEDL with your home or work insurance provider are your responsibility!*)

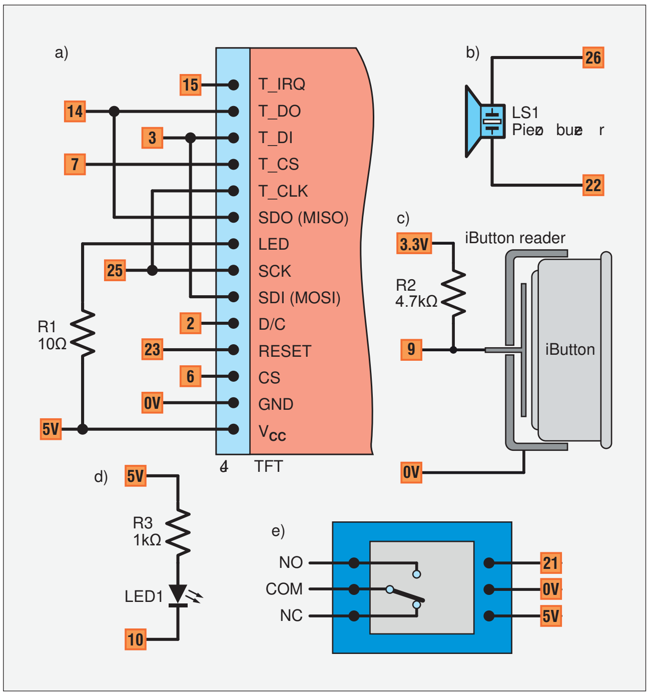


Fig.4. Pin connections between the Micromite and the five main elements of the *iEDL:* a) The TFT has 14 contacts connected as shown (note the 10Ω backlight resistor); b) the piezo is connected between pin 26 (PWM 2A) and I/O pin 22; c) The iButton reader (with 4.7kΩ pull-up) is connected between I/O pin 9, and 0V; d) An LED (and series current-limiting resistor) is connected between I/O pin 10, and 5V; e) The relay module needs 5V power. I/O pin 21 controls the change-over contact.

Circuit assembly

The most complex connection inside the *iEDL* is between the Micromite and the touchscreen. If you are using an *MKC*, *DM* and *TFT-adaptor*, then these connections are already made for you (and likewise if using a 28-pin *Backpack*). However, if you are building an *iEDL* with an Explore28 (or Explore64) module, then you will need to make the ten connections shown in Fig.4a, plus the 10Ω current-limiting resistor required for the TFT backlight. These connections ensure all aspects of the touchscreen work when using the Micromite's built-in TFT driver.

To provide audible feedback to the user, a piezo sounder is connected between pin 26 and pin 22 (see Fig.4b). Pin 26 is a PWM output, and when MMBASIC is set to output a square wave on PWM channel 2A, and I/O pin 22 is set low (or high, rather than left 'floating'), then a tone will be heard. Unless you are using an *MKC* with the *TFT adaptor* (which has a built-in piezo), you will need to connect an external piezo sounder to the Micromite.

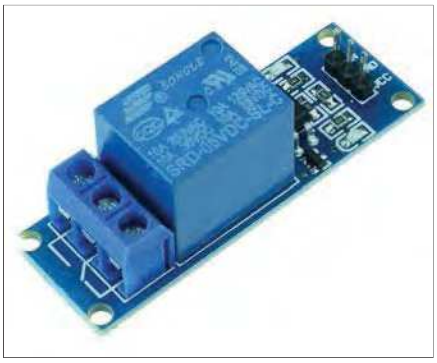


Fig.3. This low-cost relay module is readily available online. It contains a set of change-over contacts, and hence can be interfaced to the electromechanical hardware that in turn controls the locking/releasing of the door.

The iButton reader is connected to Pin 9, as shown in Fig.4c. This requires a 4.7kΩ pull-up resistor to function correctly (as explained in *MIWM Part 32*, *PE* Jan 2022).

Whether the iButton reader has a built-in LED, or you use an external LED, you will require the usual LED current-limiting resistor to be connected in series. The exact value will depend on the LED being used, but a typical value is somewhere between 330Ω and 1kΩ. This LED is useful for highlighting where the iButton reader is located in dark conditions. It is connected to pin 10, as shown in Fig.4d; setting pin 10 low turns the LED on.

Finally, the relay module is connected to Micromite pin 21 (and 5V power), as shown in Fig.4e. Setting pin 21 low turns the relay on (and hence switches the relay's change-over contacts).

Hardware assembly is straightforward whichever Micromite module you use; and it's easy to understand how the Micromite interacts with each element. Remember, it is the MMBASIC code which makes everything come to life (and enables the intelligence and features).

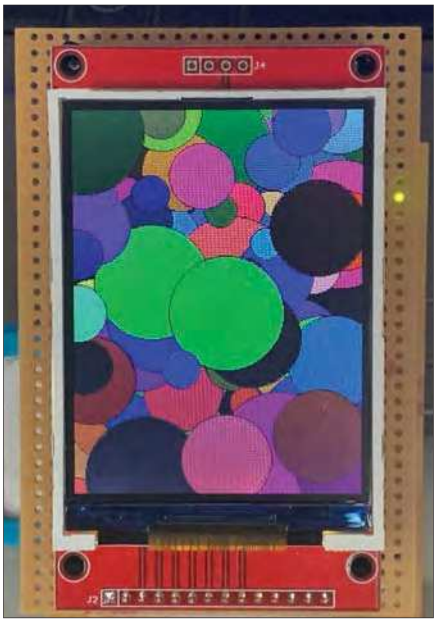


Fig.5. `GUI TEST LCDPANEL` should result in this built-in test-pattern animation appearing on the TFT screen.

Configuring the iEDL

Once you have completed the required connections between the Micromite and each *iEDL* element, perform all the usual visual checks. When everything looks correct, apply power to the *iEDL*, and then connect the Micromite to your preferred terminal program (eg, TeraTerm). When a connection is established, you will need to configure two specific Micromite `OPTION`s, followed by careful calibration of the touchscreen. So, assuming that the two `OPTION`s aren't yet set, start by configuring the TFT display driver with the command:

```
OPTION LCDPANEL ILI9341,L,2,23,6
```

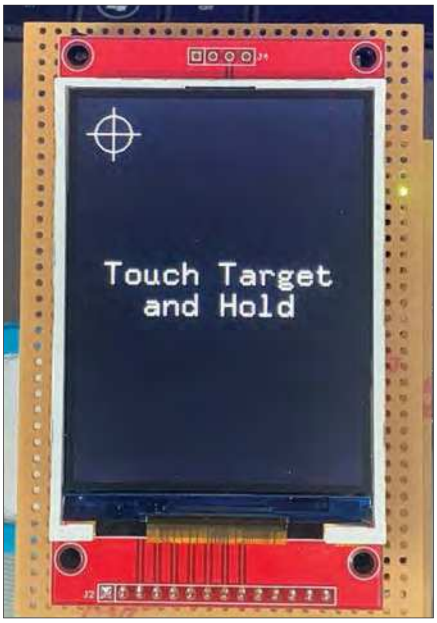


Fig.6. The touchscreen is calibrated with the command `GUI CALIBRATE`. Use a plastic stylus and simply follow the on-screen instructions to complete calibration.

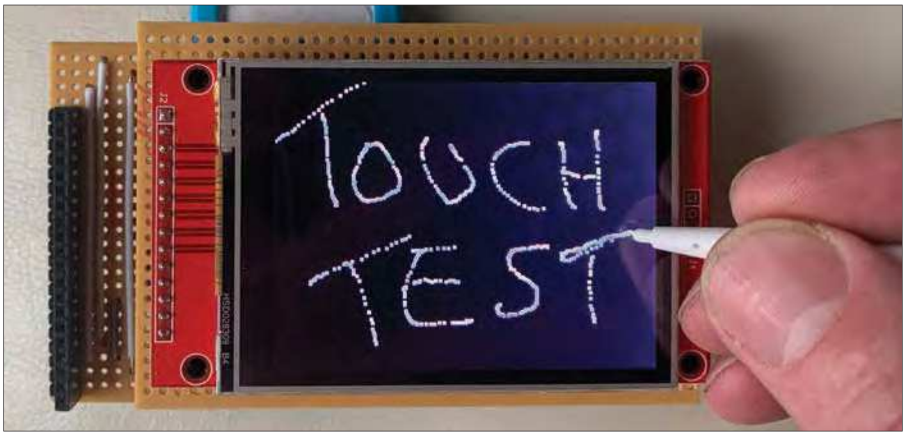


Fig.7. `GUI TEST TOUCH` allows you to test the accuracy of the calibration. If it's too inaccurate, simply use `GUI CALIBRATE` again.

On pressing `Enter`, the TFT should go blank (rather than appearing whiteish/grey). If it doesn't, then power down, recheck everything, then power back up. The TFT's backlight should flash briefly, and then the display will go blank. Now type `GUI TEST LCDPANEL` and you should see the usual built-in animated test pattern (a circles patterns – see Fig.5).

Next, you need to configure the touchscreen. This is carried out with the command `OPTION TOUCH 7,15` followed by the command `GUI CALIBRATE`

On pressing `Enter`, the TFT will display the calibration screen, as shown in Fig.6. Follow the onscreen instructions using a plastic stylus to touch the four crosshairs that pop up, one at a time. Once this step is completed, you can check the accuracy of calibration by typing the command `GUI TEST TOUCH`

On pressing `ENTER`, the screen will clear, and you can use a stylus to write words on the screen (see Fig.7). If the position of the pixels appearing are offset too far from the tip of the stylus, then you will need to recalibrate the screen (`GUI CALIBRATE`). Once accurate screen configuration has been completed you can proceed to testing the piezo sounder, the LED, the relay module and the iButton reader.

Testing the iEDL

To test the piezo sounder, begin by setting PWM channel 2A (pin 26) to output a square wave with a frequency of 1kHz. This is achieved by typing the command: `PWM 2,1000,50` (1000Hz with a 50% duty cycle). Nothing should be heard at this point since you also need to set pin 22 to 0V, which is achieved by typing the command: `SETPIN 22,DOUT`

On pressing `Enter`, you should hear the piezo sounder (if not, then check the commands have been typed correctly, and also the piezo's two pins are connected to pins 26 and 22. The tone of the sounder can be changed by altering the relevant PWM parameter. To see this (or rather to hear it), type: `PWM 2,500,50`

This will change the frequency from 1kHz to 500Hz. The piezo is silenced by typing: `SETPIN 22,OFF` (which causes pin 22 to float). Alternatively, the piezo sounder can be silenced by typing: `PWM 2,STOP`

The LED is tested by setting I/O pin 10 to a digital output. To turn it on, set pin 10 low by typing: `SETPIN 10,DOUT`

If the LED does not light, first check the command has been entered correctly, then check the connections to the series resistor, and to the LED. Be sure to check the value of the resistor is somewhere in the region between 330Ω to 1kΩ. If it is too high (possibly due to mis-reading the resistor's colour bands) then the chances are the LED could be so dim, that it appears not to be working at all. If after these checks it still doesn't light, then ensure that the LED is installed correctly (remember, LEDs are polarised components and hence they must be inserted the correct way round).

Once the LED is lit, then check it is turned off by typing the command: `PIN(10)=1`

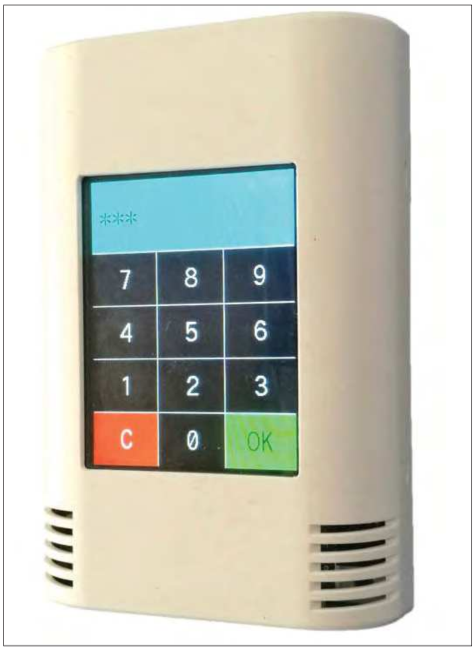
The relay module is tested in a similar way as the LED – simply replace the reference to pin 10 with pin 21. Thus, to energise the relay, type the following command: `SETPIN 21,DOUT`

You should hear the relay click; if not, perform the usual checks. Once this is working correctly, ensure you can switch the relay off again by typing: `PIN(21)=1` (again, you should hear a click as the relay coil is de-energised).

Finally, you need to test the iButton reader. To do this, you will require at least one iButton; this can be either a DS1990 (ID only), or a DS1971 (EEPROM) iButton.

Type in the following program into the `EDIT`or:

```
DO
  ONEWIRE RESET 9
  TEXT 160,120,MM.ONEWIRE,CM,1,3
  PAUSE 10
LOOP
```



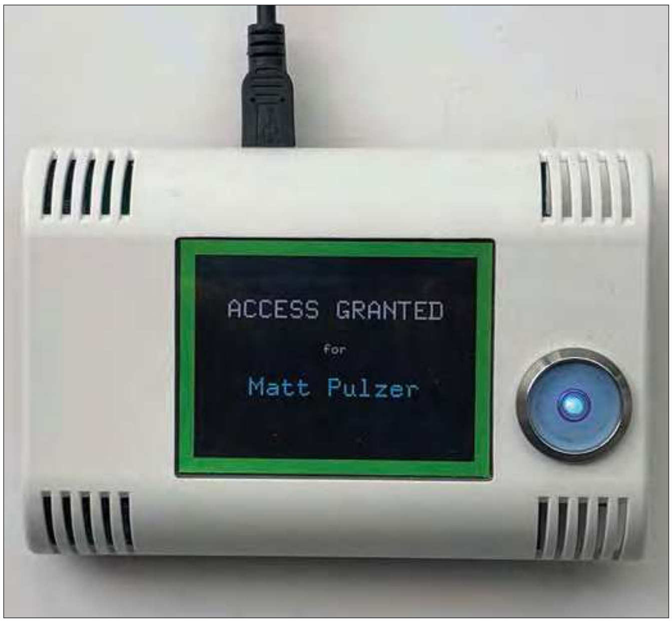


Fig.8. The AZ-Touch enclosure is an attractive wall mounted case. a) (left) the image that convinced me to try and fit an *iEDL* inside. b) (right) my first attempt at fitting the *iEDL* into a modified AZ-Touch enclosure. Note the iButton reader lower right.

Next, `RUN` the program and check that a `0` character is displayed in the centre of the TFT. Then tap (and hold) an iButton onto the reader and make sure that the `0` changes to a `1` while the iButton is in contact with the reader. If this test fails, then check the code is entered correctly, and also that the iButton reader is connected to pin 9. Also check the value of the pull-up resistor; it needs to be 4.7kΩ.

Once the above tests have been successfully completed, then we can be guaranteed that the *iEDL* has been assembled correctly. The *iEDL* is now ready for the main program code to be loaded; however, this will have to wait until next month. So, in the meantime, why not try and modify some of the recent iButton demo code so that it displays information (such as the iButton's unique ID number) on the TFT display instead of in the Terminal screen. Essentially, this means replacing the relevant `PRINT` commands with `TEXT` commands, along with using the correct parameters that the `TEXT` command requires. Refer to the simple `DO... LOOP` program listed above for such an example, and also refer to the *Micromite User Manual* to check the parameters used – for example, to add some colour.

Housing the iEDL

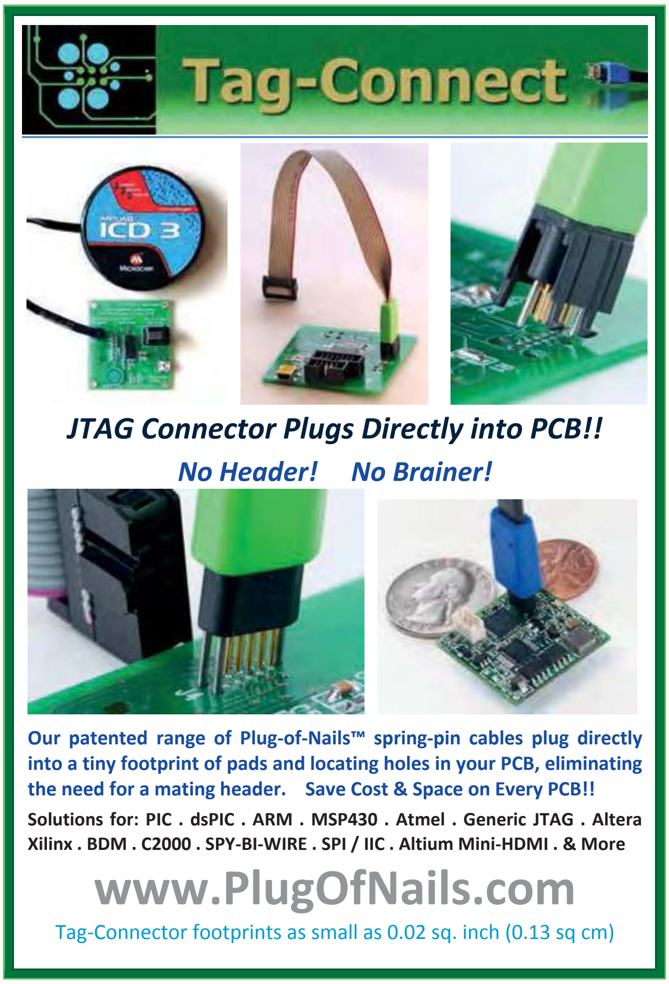
One thing to think about is how to install the *iEDL* into a suitable enclosure. Many technically brilliant electronic projects are let down by being housed in a poor casing. Custom 3D-printed enclosures have become more popular over the last few years; however, this approach is only practical if you have access to a 3D-printer (and arguably have lots of spare time to print the numerous prototypes that you'll end up printing so that you get everything positioned correctly).

However, one nice looking (and readily available) enclosure that grabbed my attention recently was the AZ-Touch (see Fig.8). This is a wall-mounted unit with a cut-out for either a 2.4-inch or 2.8-inch ILI9341 touchscreen. As a bonus, a compatible ILI9341 TFT is included in the AZ-Touch kit (Google for a list of AZ-Touch sources).

Although many pictures online show the AZ-Touch mounted in a portrait orientation, there is nothing stopping us from using this enclosure in a landscape orientation. So, I decided to purchase a handful of these kits and attempted to modify things to see if I could successfully house the *iEDL* inside. An early prototype is shown in Fig.8b, but do note that the version of the *iEDL* shown in the photo is based on an Explore64, and is positioned behind the front panel, rather than installed inside the enclosure. I just thought that I would share this first attempt at trying to fit an *iEDL* into an AZ-Touch enclosure. I hope this inspires you to think about how to house your *iEDL*.

Next time

Now that we have assembled, configured, and tested the *iEDL* mini-project, next month we will discuss how the program code works. Until then, stay safe, and have FUN!


Tag-Connect

JTAG Connector Plugs Directly into PCB!!

No Header! No Brainer!

Our patented range of Plug-of-Nails™ spring-pin cables plug directly into a tiny footprint of pads and locating holes in your PCB, eliminating the need for a mating header. Save Cost & Space on Every PCB!!

Solutions for: PIC . dsPIC . ARM . MSP430 . Atmel . Generic JTAG . Altera Xilinx . BDM . C2000 . SPY-BI-WIRE . SPI / IIC . Altium Mini-HDMI . & More

www.PlugOfNails.com

Tag-Connector footprints as small as 0.02 sq. inch (0.13 sq cm)

Practical Electronics PCB SERVICE

PROJECT	CODE	PRICE
APRIL 2022		
64-key MIDI Matrix shield	23101211	8.95
64-key MIDI Matrix shield	23101212	11.95
High-current Battery Balancer	14102211	10.95
Digital FX Unit – using potentiometer	01102211	9.95
Digital FX Unit – using BCD switch	01102212	9.95
Universal Audio PSU	AO1-APR22	11.95
MARCH 2022		
Mini Isolated Serial Link	24102211	£5.95
Busy Loo Indicator	16112201	£5.95
Analogue Vocoder – Band-pass filter board	AO1-MAR22	9.95
Analogue Vocoder – HP/LP filter board	AO2-MAR22	9.95
FEBRUARY 2022		
Arduino-based Power Supply	18106201	9.95
Battery Monitor Logger	11106201	10.95
Electronic Wind Chimes	23011201	10.95
Analogue Vocoder – Driver Amplifier	AO-FEB22	8.95
JANUARY 2022		
Vintage battery Radio Li-ion Power Supply	11111201	9.95
MiniHeart: A Miniature Heartbeat Simulator	01109201	8.95
DECEMBER 2021		
AM/FM/SW Digital Receiver	CSE200902A	13.95
Balanced Input and Attenuator for USB CODEC	01106202	11.95
NOVEMBER 2021		
Dual Battery Lifesaver	11111202	£6.95
OCTOBER 2021		
Mini Wi-Fi LCD BackPack	24106201	£8.95
SEPTEMBER 2021		
USB SuperCodec PCB	01106201	£14.95
Audio DDS Oscillator PCB	01110201	£5.95
Audio DDS Oscillator rotary encoder	01110201-ENC	6.95
Programming Adaptor Board for Audio DDS Oscillator	01110202	£5.95
High-power Ultrasonic Cleaner main PCB	04105201	£14.95
High-power Ultrasonic Cleaner front-panel PCB	04105202	
Night Keeper Lighthouse PCB	08110201	£5.95
AUGUST 2021		
Ol’ Timer PCB	19104201	£11.95
Ol’ Timer 8x8 RGB LED module using WS2812B	19104201-88	£8.95
Ol’ Timer set of acrylic case pieces and spacer	19104201-ACR	£8.75
Ol’ Timer DS3231 RTC IC wide SOIC-16	19104201-RTC	£5.95
Wideband Digital RF Power Meter	04106201	£9.75
Switchmode 78xx regulators ***(PACK of 5!)***	18105201	£7.95
Cool Beans SMAD display	CB-AUG21	£11.95
JULY 2021		
ATtiny816 Breakout / Dev Board with Capacitive Touch	24110181	£9.75
IR Remote Control Assistant (Jaycar version)	15005201	£8.95
IR Remote Control Assistant (Altronics version)	15005202	£8.95
PIC18F Development Board	PNM-JUL21	£12.95
Microphone Preamplifier	AO-JUL21	£11.95
JUNE 2021		
Roadies’ Test Signal Generator (surface-mount version)	01005201	£8.95
Roadies’ Test Signal Generator (through-hole version)	01005202	£9.95
Touchscreen Wide-range RCL Box (Resistor module)	04104201	£18.95
Touchscreen Wide-range RCL Box (Ind/Cap module)	04104202	
KickStart Part 3 – Gyrator-based Audio Filter	KS3-2021	£7.95

PROJECT	CODE	PRICE
MAY 2021		
7-Band Equaliser (Mono)	01104201	£8.95
7-Band Equaliser (Stereo)	01104202	£10.95
Car Altimeter	05105201	£7.95
APRIL 2021		
Reflow Oven – DSP Active Crossover (CPU)	01106193	£19.95
Reflow Oven – DSP Active Crossover (Front panel)	01106195	
Reflow Oven – DSP Active Crossover (LCD)	01106196	
Frequency Reference Signal Distributor	CSE200103	£8.95
MARCH 2021		
Nutube Guitar Effects Pedal	01102201	£12.95
Programmable Thermal Regulator (Peltier Interface)	21109181	£18.95
Programmable Thermal Regulator (Peltier Driver)	21109182	
Tunable HF Preamp	CSE190502	£8.95
FEBRUARY 2021		
4G Remote Monitoring	27111191	£9.95
JANUARY 2021		
Nutube Valve Preamplifier	01112191	£12.95
Arduino DCC Controller	09207181	£10.95
DECEMBER 2020		
Pseudo-Random Sequence Generator	16106191	£7.95
Clever Charger	14107191	£11.95
PE Theremin Amplifier	AO-1220-01	£8.95
NOVEMBER 2020		
LED Christmas Tree (1 off)	16107181-1	£6.95
LED Christmas Tree (4 off)	16107181-2	£14.95
LED Christmas Tree (12 off)	16107181-3	£24.95
LED Christmas Tree (20 off)	16107181-4	£34.95
USB/SPI Interface Board	16107182	£5.95
45V/8A Power Supply PCB *plus* acrylic spacer	18111181	£14.95
45V/8A Power Supply front panel five-way display bezel	18111181-BZ	£3.95
Five-way LCD Panel Meter/Display	18111182	£7.95
OCTOBER 2020		
Digital Audio Millivoltmeter	04108191	£9.95
Precision Signal Amplifier	04107191	£7.95
SEPTEMBER 2020		
PE Theremin PSU	AO-0920-01	£5.95
PE Theremin PSU transformer	AO-0920-02	£7.95
Micromite Explore-28	07108191	£6.95
Ultrabrite LED Driver	16109191	£6.95
AUGUST 2020		
Micromite LCD BackPack V3	07106191	£9.95
Steering Wheel Audio Button to Infrared Adaptor	05105191	£7.95
JULY 2020		
AM/FM/CW Scanning HF/VHF RF Signal Generator	04106191	£13.95
Speech Synthesiser with the Raspberry Pi Zero	01106191	£5.95
PE Mini-organ PCB	AO-0720-01	£14.95
PE Mini-organ selected parts	AO-0720-02	£8.95
High-current Solid-state 12V Battery Isolator – control	05106191	£6.95
High-current Solid-state 12V Battery Isolator FET (2oz)	05106192	£9.95
JUNE 2020		
Arduino breakout board – 3.5-inch LCD Display	24111181	£6.95
Six-input Audio Selector main board	01110191	£10.95
Six-input Audio Selector switch panel board	01110192	

PCBs for most recent *PE*/*EPE* constructional projects are available. From the July 2013 issue onwards, PCBs with eight-digit codes have silk screen overlays and, where applicable, are double-sided, have plated-through holes, and solder mask. They are similar to photos in the project articles. Earlier PCBs are likely to be more basic and may not include silk screen overlay, be single-sided, lack plated-through holes and solder mask.

Always check price and availability in the latest issue or online. A large number of older boards are listed for ordering on our website.

In most cases we do not supply kits or components for our projects. For older projects it is important to check the availability of all components before purchasing PCBs.

Back issues of articles are available – see Back Issues page for details.

CLASSIFIED ADVERTISING

Practical Electronics

If you want your advertisements to be seen by the largest readership at the most economical price then our classified page offers excellent value. The rate for semi-display space is £10 (+VAT) per centimetre high, with a minimum height of 2·5cm. All semi-display adverts have a width of 5.5cm. The prepaid rate for classified adverts is 40p (+VAT) per word (minimum 12 words).

Cheques are made payable to 'Practical Electronics'. **VAT must be added**. Advertisements with remittance should be sent to: Practical Electronics, 113 Lynwood Drive, Wimborne, Dorset, BH21 1UU.

Tel 07973518682 Email: pe@electronpublishing.com

For rates and further information on display and classified advertising please contact our Advertisement Manager, Matt Pulzer – see below.

Practical Electronics reaches more UK readers than any other UK monthly hobby electronics magazine. Our sales figures prove it.

We have been the leading monthly magazine in this market for the last twenty-seven years.

BOWOOD ELECTRONICS LTD

Suppliers of Electronic Components

www.bowood-electronics.co.uk

Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield,
Derbyshire S40 2QR. Sales: 01246 200 222

Send large letter stamp for Catalogue

Electrical Industries Charity (EIC)
We help people working in the electrical, electronics and energy community as well as their family members and retirees.

We use workplace programmes that give the industry access to financial grants and a comprehensive range of free and confidential services.
www.electricalcharity.org

electrical industries charity

COAST ELECTRONICS

BREAKOUTS-COMPONENTS-
CONTRACT DESIGN-3D PRINTER PARTS-
MUSICAL-MICROCONTROLLERS
WWW.COASTELECTRONICS.CO.UK

Andrew Kenny – Qualified Patent Agent
EPO UKIPO USPTO
Circuits Electric Machinery Mechatronics
Web: www.akennypatentm.com
Email: Enquiries@akennypatentm.com
Tel: 0789 606 9725

MISCELLANEOUS

VALVES AND ALLIED COMPONENTS IN STOCK. Phone for free list. Valves, books and magazines wanted. Geoff Davies (Radio), tel. 01788 574774.

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits.
Visit www.cstech.co.uk

ADVERTISING INDEX

BRUNNING SOFTWARE	37
CRICKLEWOOD ELECTRONICS	45
ESR ELECTRONIC COMPONENTS	61
FLOWCODE	9
MATRIX RECRUITMENT	25
HAMMOND ELECTRONICS Ltd	10
JPG ELECTRONICS	72
MICROCHIP	Cover (ii)
PEAK ELECTRONIC DESIGN	Cover (iv)
PICO TECHNOLOGY	5
POLABS D.O.O.	65
QUASAR ELECTRONICS	3
SILICON CHIP	6
STEWART OF READING	61
TAG-CONNECT	69

Advertisement offices

Matt Pulzer

Electron Publishing Ltd
1 Buckingham Road
Brighton
East Sussex BN1 3RA

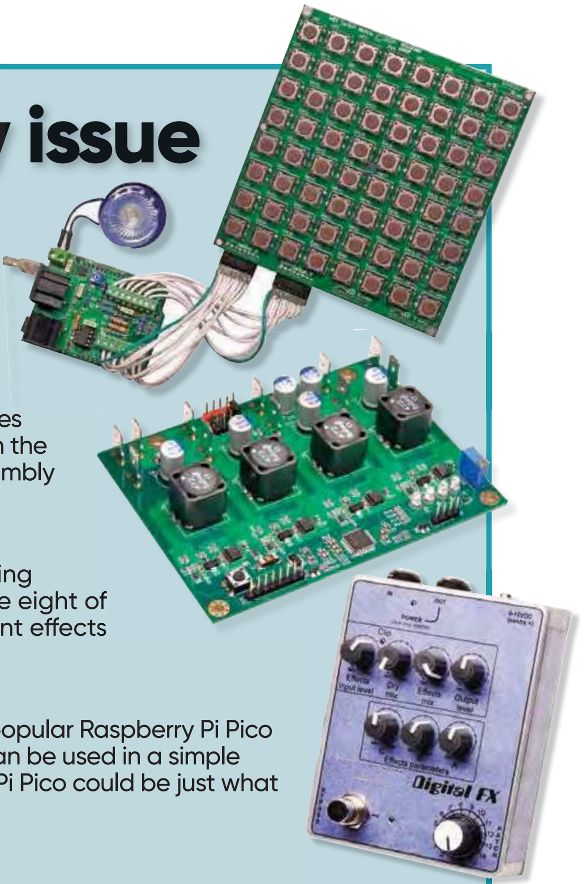
Tel 07973 518682

Email pe@electronpublishing.com

Web www.electronpublishing.com

For editorial contact details see page 7.

Next Month – in the May issue



64-key Arduino MIDI matrix – Part 2
We've described the low-cost hardware that you can build to work with MIDI, including a comprehensive MIDI Encoder Shield and a MIDI Key Matrix to drive it. Next, we look at some software to make even better use of this hardware, and we'll show you how you can use it with Android smartphones and tablets.

High-current Battery Balancer – Part 2
Our new High Current Battery Balancer is an advanced design which provides high efficiency and fast balancing by efficiently transferring charge between the connected cells or batteries. This second and final article describes the assembly and testing steps, and how to use it.

Digital FX Unit – Part 2
Our new DFX unit can produce 15 different effects for musicians or recording technicians to customise their instruments and sounds. You can customise eight of those effects – and this article describes how to create and install different effects patches into the unit's EEPROM.

KickStart – Part 8: Introducing the Raspberry Pi Pico
This eighth instalment of KickStart provides you with an introduction to the popular Raspberry Pi Pico microcontroller. We will describe the board's architecture and show how it can be used in a simple application. If you've not dabbled in microcontrollers before, the Raspberry Pi Pico could be just what you've been waiting for!

PLUS!
All your favourite regular columns from *Audio Out, Cool Beans and Circuit Surgery*, to *Make it with Micromite* and *Net Work*.

On sale 7 April 2022

Content may be subject to change

WELCOME TO JPG ELECTRONICS
Selling Electronics in Chesterfield for 29 Years

Open Monday to Friday 9am to 5:30pm
And Saturday 9:30am to 5pm

- Aerials, Satellite Dishes & LCD Brackets
- Audio Adaptors, Connectors & Leads
- BT, Broadband, Network & USB Leads
- Computer Memory, Hard Drives & Parts
- DJ Equipment, Lighting & Supplies
- Extensive Electronic Components
  - *ICs, Project Boxes, Relays & Resistors*
- Raspberry Pi & Arduino Products
- Replacement Laptop Power Supplies
- Batteries, Fuses, Glue, Tools & Lots more...

T: 01246 211 202
E: sales@jpgelectronics.com
JPG Electronics, Shaw's Row,
Old Road, Chesterfield, S40 2RB
W: www.jpgelectronics.com

JPG Electronics

Retail & Trade Welcome • Free Parking • Google St View Tour: S40 2RB

NEW subscriptions hotline!

Practical Electronics

We have changed the way we sell and renew subscriptions. We now use 'Select Publisher Services' for all print subscriptions – to start a new subscription or renew an existing one you have three choices:

1. Call our NEW print subscription hotline: 01202 087631, or email: pesubs@selectps.com
2. Visit our shop at: www.electronpublishing.com
3. Send a cheque (payable to: 'Practical Electronics') with your details to:

   Practical Electronics Subscriptions, PO Box 6337,
   Bournemouth BH1 9EH, United Kingdom

Remember, we print the date of the last issue of your current subscription in a box on the address sheet that comes with your copy.

Digital subscribers, please call 01202 880299 or visit: www.electronpublishing.com

Published on approximately the first Thursday of each month by Electron Publishing Limited, 1 Buckingham Road, Brighton, East Sussex BN1 3RA. Printed in England by Acorn Web Offset Ltd., Normanton WF6 1TW. Distributed by Seymour, 86 Newman St., London W1T 3EX. Subscriptions UK: £29.99 (6 months); £54.99 (12 months); £104.99 (2 years). EUROPE: airmail service, £33.99 (6 months); £63.99 (12 months); £119.99 (2 years). REST OF THE WORLD: airmail service, £41.99 (6 months); £77.99 (12 months); £149.99 (2 years). Payments payable to 'Practical Electronics', Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom. Email: pesubs@selectps.com. PRACTICAL ELECTRONICS is sold subject to the following conditions, namely that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price shown on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter whatsoever.

Did you know our online shop now sells the current issue of *PE* for £5.49 inc. p&p?

Practical Electronics
The UK's premier electronics and computing maker magazine
Circuit Surgery Exploring the LM35 temp sensor
Audio Out Vocoder: Audio PSU
Cool Beans Mastering switch debounce
Make it with Micromite Build an iButton-based Electronic Door Lock
64-key MIDI Matrix
ARDUINO
High-current Battery Balancer
WIN! Microchip MPLAB ICD 4 In-Circuit Debugger WIN
Digital FX Unit
Learn Flowcode Programming: PIC, Arduino and 16x2 LCD
PLUS!
Techno Talk – Go eco, get ethical!
Fox Report – Another fine mess: moving to Windows 11
Net Work – Scanners, eVTOLs and the latest from space
Apr 2022 £5.49
www.electronpublishing.com
@practicalelec
practicalelectronics
Make it with Micromite Using and interfacing an EEPROM iButton
WIN! Microchip RN2903 LoRa technology Mote WIN!
Mini Isolated Serial Link
Mar 2022 £5.49

You read that right! We now sell the current issue of your favourite electronics magazine for exactly the same price as in the High Street, but we deliver it straight to your door – and for UK addresses we pay the postage. No need to journey into town to queue outside the newsagent. Just go to our website, set up an account in 30 seconds, order your magazine and we'll do the rest.

www.electronpublishing.com

PEAK®
electronic design ltd

15% discount!

Use code *PE15* during checkout

limited time only!

ZEN50 Zener Diode Analyser (inc. LEDs, TVSs etc)

Popular product!

The ***Atlas ZEN*** (model ZEN50) is ideal for testing Zeners, VDRs, transient suppressors, LED strings and many other components.

- **Measure Zener Voltage (from 0.00 up to 50.00V!)**
- **Measure Slope Resistance.**
- **Selectable test current: 2mA, 5mA, 10mA and 15mA.**
- **Very low duty cycle to minimise temperature rise.**
- **Continuous measurements.**
- **Single AAA battery (included) with very long battery life.**
- **Gold plated croc clips included.**
- **Can measure forward voltage of LEDs and LED strings too.**

~~£46.50~~ £39.53
~~£38.75+VAT~~ with discount!

LCR45 LCR and Impedance Meter with Auto and Manual modes

Advanced firmware!

The powerful LCR45 meter that not only identifies and measures your passive components (Inductors, Capacitors and Resistors) but also measures complex impedance, magnitude of impedance with phase and admittance too! Auto and Manual test modes allow you to specify the test frequency and component type.

- **Continuous fluid measurements.**
- **Improved measurement resolution: (<0.2µH, <0.2pF).**
- **Test frequencies: DC, 1kHz, 15kHz, 200kHz.**
- **Measure the true impedance of speakers and more.**
- **Great for hobbyists and professionals.**

~~£91.50~~ £77.78
~~£76.25+VAT~~ with discount!

DCA55 Semiconductor Analyser - Identify your semi's

Now with backlit display and AAA battery!

Connect any way round to identify the type of component and the pinout! Also measures many parameters including transistor gain, base-emitter voltages, MOSFET thresholds, LED voltages etc. Complete with a comprehensive illustrated user guide. Includes an Alkaline battery so you're ready to go straight away.

- **Transistors (including NPN/PNP, darlingtons, Si & Ge).**
- **Measure hFE, V_{BE} and leakage.**
- **Diodes and LEDs. Measure V_F.**
- **MOSFETs. Measure $V_{GS(th)}$.**
- **Gold plated hook probes.**
- **Long battery life.**
- **Free technical support for life.**
- **Comprehensive instruction book.**
- **2 year warranty.**

atlas DCA
model DCA55
AAA Battery Edition
PEAK electronic design ltd
Base-Emitter
VBE=1.039V
semiconductor component analyser
automatic lead identification
on - test
scroll - off
warning: avoid charge/voltages

~~£52.50~~ £44.63
~~£43.75+VAT~~ with discount!

ESR70 Equivalent Series Resistance and Capacitance Meter

Now with backlit display and AAA battery!

The new improved ESR70 is great for identifying capacitors (especially electrolytics) that are suffering from high ESR. ESR is a great indicator of capacitor condition, especially in applications demanding high ripple currents such as reservoir capacitors, SMPSUs, motor capacitors etc. Includes Alkaline AAA battery and a very comprehensive user guide. The ESR70 will even perform a controlled discharge of your caps!

atlas ESR+
model ESR70
PEAK electronic design ltd
Cap = 22.28µF
ESR = 0.48Ω
ESR, low resistance and capacitance meter
+ Audible Alerts
with automatic discharge function
on - test
off

~~£84.00~~ £71.40
~~£70.00+VAT~~ with discount!

It's only possible to show summary specifications here. Please ask if you'd like detailed data. Further information is also available on our website. Product price refunded if you're not happy.

www.peakelec.co.uk
sales@peakelec.co.uk

Atlas House, 2 Kiln Lane, Harpur Hill Business Park
Buxton, Derbyshire, SK17 9JL, UK. Tel: 01298 70012